Stochastic Analysis and Learning-based Algorithms for Resource Provisioning in Optical Networks

Von der Fakultät für Elektrotechnik, Informationstechnik, Physik
der Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades eines Doktors

der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von:	Sandeep Kumar Singh
aus:	Sukhanagar, India
eingereicht am:	25 März 2019
mündliche Prüfung am:	14 Juni 2019

1. Referentin:	Admela Jukan, Professor, TU Braunschweig
2. Referent:	Vincent W. S. Chan, Professor, MIT
3. Referent:	Krishna M. Sivalingam, Professor, IIT Madras

Druckjahr: 2019

Abstract

The unprecedented growth in Internet traffic has driven the innovations in provisioning of optical resources as per the need of bandwidth demands such that the resource utilization and spectrum efficiency could be maximized. With the advent of the next generation flexible optical transponders and switches, the flexible-grid-based elastic optical network (EON) is foreseen as an alternative to the widely deployed fixed-grid-based wavelength division multiplexing networks. At the same time, the flexible resource provisioning also raises new challenges for EONs. One such challenge is the spectrum fragmentation. As network traffic varies over time, spectrum gets fragmented due to the setting up and tearing down of non-uniform bandwidth requests over aligned (i.e., continuous) and adjacent (i.e., contiguous) spectrum slices, which leads to a non-optimal spectrum allocation, and generally results in higher blocking probability and lower spectrum utilization in EONs. To address this issue, the allocation and reallocation of optical resources are required to be modeled accurately, and managed efficiently and intelligently.

The modeling of routing and spectrum allocation in EONs with the spectrum contiguity and spectrum continuity constraints is well-investigated, but existing models do not consider the fragmentation issue resulted by these constraints and non-uniform bandwidth demands. This thesis addresses this issue and considers both the constraints to computing exact blocking probabilities in EONs with and without spectrum conversion, and with spectrum reallocation (known as defragmentation) for the first time using the Markovian approach. As the exact network models are not scalable with respect to the network size and capacity, this thesis proposes load-independent and load-dependent approximate models to compute approximate blocking probabilities in EONs. Results show that the connection blocking due to fragmentation can be reduced by using a spectrum conversion or a defragmentation approach, but it can not be eliminated in a mesh network topology.

This thesis also deals with the important network resource pro-

visioning task in EONs. To this end, it first presents algorithmic solutions to efficiently allocate and reallocate spectrum resources using the fragmentation factor along spectral, time, and spatial dimensions. Furthermore, this thesis highlights the role of machine learning techniques in alleviating issues in static provisioning of optical resources, and presents two use-cases: handling time-varying traffic in optical data center networks, and reducing energy consumption and allocating spectrum proportionately to traffic classes in fiber-wireless networks.

Kurzfassung

Das beispiellose Wachstum des Internetverkehrs hat zufolge, dass die Innovationen für die Bereitstellung der optischen Ressourcen je nach Bandbreitenbedarf vorangetrieben werden. Das Hauptziel ist dabei die optische Ressource und das Spektrum effizienter zu nutzen. Mit der Einführung von den flexiblen optischen Transpondern und Switches der nächsten Generation stellen die Elastischen Optischen Netze (EON) mit einer flexiblen Zuweisung des Spektrums eine Alternative zu den weitverbreiteten Wellenlängenmultiplexnetzwerken mit den fest definierten Größen des genutzten Spektrums dar. Gleichzeitig bringt die flexible Nutzung des Spektrums in EONs auch neue Herausforderungen mit sich. Eine von diesen Herausforderungen ist die Fragmentierung des Spektrums. Diese Fragmentierung entsteht dadurch, dass die Netzwerkverkehrslast sich im Laufe der Zeit ändert und so wird das Spektrum aufgrund des Verbindungsaufbaus und -abbaus fragmentiert. Die Größe der Spektrumsfragmente ist von den vorherigen Reservierungen des Spektrums abhängig. Das für eine Verbindung notwendige Spektrum wird durch aufeinander folgende (kontinuierliche) und benachbarte (zusammenhängende) Spektrumsabschnitte (Slots) gebildet. Jede Verbindung braucht gewöhnlich eine unterschiedlich große Bandbreite und somit eine unterschiedliche Anzahl von Slots. Dies führt nach den zahlreichen Reservierungen und Freisetzungen des Spektrums zu einer nicht optimalen Zuordnung, die in einer höheren Blockierungs-wahrscheinlichkeit der neuen Verbindungsanfragen und einer geringeren Auslastung von EONs resultiert. Um dieses Problem zu lösen, müssen die Zuweisung und Neuzuordnung des Spektrums in EONs genau modelliert und effizient sowie intelligent verwaltet werden.

Die Modellierung von Routing und Spektrumzuweisung in EONs mit den Einschränkungen der Spektrumskontiguität und Spektrumskontinuität ist bereits ausreichend untersucht. Allerdings berücksichtigen die bestehenden Modelle nicht das Fragmentierungsproblem, das durch diese Einschränkungen (Kontiguität und Kontinuität) und keine einheitliche Bandbreitennutzung entsteht. Die-

se Arbeit beschäftigt sich mit dem Fragmentierungsproblem und berücksichtigt dabei die beiden obengenannten Einschränkungen. Unter diesen Annahmen wurden analytische Modelle zur Berechnung einer exakten Blockierungswahrscheinlichkeit in EONs mit und ohne Spektrumskonvertierung erarbeitet. Außerdem umfasst diese Arbeit eine Analyse der Blockierungswahrscheinlichkeit im Falle einer Neuzuordnung des Sprektrums (Defragmentierung). Diese Blockierungsanalyse wird zum ersten Mal mit Hilfe der Markov-Modelle durchgeführt. Da die exakten analytischen Modelle hinsichtlich der Netzwerkgröße und -kapazität nicht skalierbar sind, werden in dieser Dissertation verkehrslastunabhängige und verkehrslastabhängige Approximationsmodelle vorgestellt. Diese Modelle bieten eine Näherung der Blockierungswahrscheinlichkeiten in EONs. Die Ergebnisse zeigen, dass die Blockierungswahrscheinlichkeit einer Verbindung aufgrund von einer Fragmentierung des Spektrums durch die Verwendung einer Spektrumkonvertierung oder eines Defragmentierungsverfahrens verringert werden kann.

Eine effiziente Bereitstellung der optischen Netzwerkressourcen ist eine wichtige Aufgabe von EONs. Deswegen befasst sich diese Arbeit mit algorithmischen Lösungen, die Spektrumressource mithilfe des Fragmentierungsfaktors von Spektral-, Zeit- und räumlichen Dimension effizient zuweisen und neu zuordnen. Darüber hinaus wird die Rolle des maschinellen Lernens (ML) für eine verbesserte Bereitstellung der optischen Ressourcen untersucht und das ML basierte Verfahren mit der statischen Ressourcenzuweisung verglichen. Dabei werden zwei Anwendungsbeispiele vorgestellt und analysiert: der Umgang mit einer zeitveränderlichen Verkehrslast in optischen Rechenzentrumsnetzen, und eine Verringerung des Energieverbrauchs und die Zuweisung des Spektrums proportional zu Verkehrsklassen in kombinierten Glasfaser-Funknetzwerken.

Acknowledgments

I would like to first quote a Sanskrit verse: *Guru Brahma Guru Vishnu Guru Devo Maheshwara, Guru Sakshat Param Brahma Tasmai Shri Gurave Namah*, which means Guru is verily the representative of the divine almighty, I salute such a Guru. Prof. Admela Jukan has been my Guru, mentor, and inspiration during the course of my doctorate study, for whom any amount of words would not be sufficient to express my gratitude. I was very fortunate to have worked with Prof. Admela Jukan, whose guidance, advice and support were remarkable. She also gave me the freedom and resources to lead independent research activities, and for that I shall be forever grateful. I would also like to thank the members of my thesis committee, Prof. Vincent Chan and Prof. Krishna Sivalingam, who kindly accepted to be my examiners, for their valuable time to evaluate my work. I am also grateful to Prof. Thomas Schneider who generously accepted to chair the examination committee.

This thesis would not have been possible without the support of wonderful people as colleagues. First of all, my special thanks goes to Dr. Wolfgang Bziuk, who has been a great researcher, and taught me the art of writing theoretical papers. I am very grateful to my colleagues Anna Engelmann, and Fransico Carpio, who were very supportive and helped me with my work. I would also like to thank Dr. Tamal Das, Dr. Marcel Caria, Jasenka Dizdarević, Mounir Bensalem, and Cao Vien Phung, for being wonderful colleagues. Special thanks also goes to our administrative staff, Ms. Christiane Geller, and Mr. Marc Schmidtchen for their support during the course of my study. Many thanks to my former professors at IIT Madras for the remarkable learning experience during my Masters study.

None of this would have been possible without the love and support of my family and friends. My parents, in-laws and sisters have always been a constant pillar of support throughout this endeavor. I could not have finished this without the support and love of my wife Nandini. I shall be forever grateful to all who have helped me and encouraged me during my study, thank you very much.

Contents

List of Figures

List of Tables

1

Introduction

The continuous growth in Internet traffic and cloud-based applications led to the rethinking of whether traditional optical backbone networks based on fixed-grid Dense Wavelength Division Multiplexing (DWDM) would be able to setup optical channels (also known as lightpaths) to fulfil emerging services requiring up to 1 Terabit/s (Tb/s) bandwidth. The widely deployed DWDM solutions indeed worked very well from its adaptation in mid 1990s until last decade in the sense that it can offer data rate per optical channel in tune of 100 Gb/s. However, for the coming decade there was a need for the technological innovations and improving spectral efficiency of optical transmission in order to solve the capacity crunch of fiber-based networks [1].

The existing (D)WDM optical transmission systems divide the low attenuation fiber spectrum bandwidth in c-band (4.475 Tera Hertz) into discrete bands, spaced usually by 50 or 100 GHz as defined by the International Telecommunication Union-Telecom (ITU-T). An optical transponder sets up an individual wavelength for carrying aggregated client demands in the DWDM transmission system. Thus, the DWDM transponders offer a fix bit rate, for example 10 Gbit/s or 40 Gbit/s, and the spectral width of each wavelength signal can not be extended beyond the ITU-T defined grid (e.g., 50 GHz) due

to the deployed wavelength sensitive DWDM devices, such as optical demultiplexers and Reconfigurable Optical Add Drop Multiplexers (ROADMs) that would filter out the spectrum beyond grid boundaries. Therefore, a wavelength channel with a bit rate of 10 Gb/s assigned to a smaller demand (e.g., 1 Gb/s) would waste the residual bandwidth, and any transponders with high bit rate, e.g., 400 Gb/s and above, have a very broad spectrum to fit into this grid. This results in highly inefficient use of the fiber capacity. Although higher modulation formats, such as dual polarized 16-Quadrature Amplitude Modulation (QAM), can be used to carry large demands with the defined grid, however the optical reach (transmission distance) reduces with higher modulations.

With the advent of the next generation flexible optical transponders, such as sliceable bandwidth variable transponders [2], the flexible-grid-based Elastic Optical Network (EON) is going to replace sooner than later the fixed-grid-based WDM networks. The reason is that instead of a single carrier technology, multi-carrier technologies, such as optical Orthogonal Frequency Division Multiplexing (OFDM) or Nyquist-Wavelength Division Multiplexing (WDM) based transponders, together with the bandwidth-variable wavelength selective switches can be used in EONs to dynamically allocate spectrum to meet the demands of diverse applications [3,4]. Additionally, these transponders can be used to support a sub-channel or a super-channel data service applications by allocating *just enough* number of the subcarriers . Unlike the traditional fixed-grid (50 GHz)-based ITU-T standard used in WDM networks, in EONs, the optical spectrum in C-band can be divided into smaller grids (e.g., 6.25 or 12.5 GHz) to accommodate low and high bit rate client demands, where the grid granularity is also known as “slot width granularity” or simply slice in ITU-T G.694.1 [4]. Additionally, EONs support distance-adaptive modulation schemes, where each subcarrier can support a fix or variable bit rates based on the modulation

formats used for a particular transmission distance. Fig. 1.1 shows the spectrum saving in a flexi-grid (50 GHz) system with different data rate signals, as compared to a fixed-grid (50 GHz) system. In

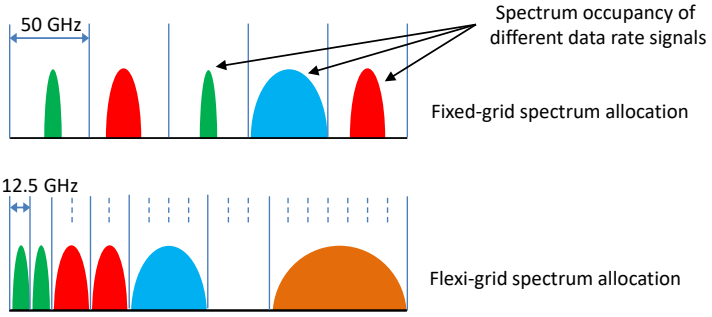

Figure 1.1.: Spectrum occupancy of different data rate signals in a fixed-grid (50 GHz) and a flexi-grid (12.5 GHz) systems.

addition to the flexible architecture, innovations in fiber technology has been brought into the limelight recently to increase the capacity limit. The space-division multiplexing (SDM) – realized by multicore or multimode fibers in EONs – has emerged as a viable networking solution for offering flexible and large bandwidth [5].

The obvious gain of flexible resource provisioning in various dimensions also brings new challenges in managing resources especially the spectrum, transponders, switches, etc. in EONs [6, 7]. In the dynamic connection set up and termination scenario, heterogeneous demands leave the gaps between occupied spectrum slices, which is called spectrum fragmentation. When the spectrum is in fragmentation state, even though sufficient, but scattered, spectrum maybe available in the network, a connection request maybe blocked if there is no required number of continuous and contiguous slices available for a new bandwidth demand. The spectrum continuity

and spectrum contiguity are in fact two fundamental constraints in setting up a lightpath in EONs. The *spectrum continuity constraint* requires an incoming connection (lightpath) request to be provisioned all-optically over the same set of subcarrier slices in all links it traverses. The constraint called *spectrum contiguity constraint* means that a connection request demanding multiple subcarriers needs to be allocated over adjacent frequency slices. These two constraints along with the resulting spectrum fragmentation have been subject to much research in optimal Routing and Spectrum Allocation (RSA) schemes, presenting an NP-hard problem [8].

Similar to the WDM, where the routing and wavelength assignment problem is usually solved by dividing it into two parts, the analogous RSA problem in EONs is solved into two sub-problems – a routing problem and a spectrum allocation problem. The routing over fiber-level paths is generally solved offline to reduce the computation complexity of the overall solution of an RSA scheme. For the spectrum allocation problem in EONs, a connection request with bandwidth demand d_k slices are allocated d_k contiguous and continuous free slices from the usable fiber spectrum in the operational lower attenuation band (typically c-band with total capacity, say C slices) on one of its fiber-level routes. There have been significant amount of works for the RSA solution: shortest path routing, κ-shortest path routing, least-loaded routing, fragmentation-aware routing, crosstalk-aware routing etc. in WDM networks as well as in EONs. The simple κ-shortest path routing, where $\kappa \geqslant 2$ routes per Origin Destination (OD) node-pair are used, has been shown to be beneficial in reducing connection blocking probability as compare to the shortest path routing ($\kappa = 1$) in optical networks [9]. Furthermore, the benefit of traffic splitting over multiple parallel routes between an OD pair has also been looked into [10, 11, 12, 13], and surveyed in [14]. Additionally, numerous studies, including some of the recent ones, for example, [15, 16, 17, 18] are dedicated

to more complex routing schemes, considering the least-utilized routes, fragmentation-aware routes, distance-adaptive modulation-aware routes, and crosstalk- or impairment-aware routes, to achieve one or more objectives, such as load balancing, lower fragmentation, lower inter-channel crosstalk etc.

Fragmentation is a well-known phenomenon in computer memory management, where a file system allocates and deallocates memory segments to contents dynamically, and therefore it has been widely investigated [19]. The most common practice is to allocate memory such that the largest free contiguous segment is intact. However, dynamic (de)allocation would still fragment the memory, so the system needs to regularly defragment the memory– using a disk defragmentor. Both methods have been adopted in the optical spectrum domain: (i) a fragmentation-aware RSA is used to minimize the fragmentation in first place (i.e., the spectrum allocation phase), and (ii) a regular defragmentation of spectrum resources further helps in consolidating scattered spectrum and thus increasing spectrum utilization.

The spectrum fragmentation issue also occurred in WDM networks due to the lone wavelength continuity constraint in establishing a lightpath over the same wavelength on all traversing links of the requested connection's path. Generally, due to the spectrum contiguity constraint even an optimal wavelength (re)allocation scheme is not always able to defragment the spectrum in order to accept an incoming connection request (as shown in chapter 3 in this thesis). The fragmentation issue is further aggravated in the EONs with the spectrum contiguity constraint, which is in addition to the legacy spectrum (wavelength) continuity constraint. Thus, connection blocking due to spectrum fragmentation is inevitable in EONs, especially in a network topology which is not linear, e.g., ring, mesh. For this reason and for network dimensioning purposes, the connection blocking analysis in elastic optical networks has been studied in

a few notable works, including [20, 21, 22]. Most of the work previously reported, starts with an analysis of blocking in a single optical link, – which is due to complexity, building it up to a computation of blocking in a network. The commonly utilized method in analyzing blocking probability in EONs is the Markov chain-based models. Since the complexity of the exact Markov models grows exponentially with the link capacity, number of links, routes, and classes of demands, implementing an approximation that is tractable, yet sufficiently accurate has been studied in the past, and was shown not to be a trivial task either. It is also analytically hard to taking into account the effect of both spectrum continuity and contiguity constraints and fragmentation while deriving the connection setup rate, i.e., the effective arrival rate at which a link allows connections to be setup in a given spectrum occupancy state. The reason is that a given number of occupied slices could be represented by fragmented as well as non-fragmented spectrum patterns, and the fragmented states could not accept an incoming connection request. The estimation of the connection setup rates taking into account the fraction of time a link stays in non-blocking states can be obtained by monitoring the link state occupancy over a long period of time, as shown by simulations by Reyes *et al.* [23]. In absence of such monitoring information, the progress towards a tractable and close to accurate approximate blocking analysis has been challenging, and slow. This thesis advances the state -of-the-art in computing connection blocking probability in EONs, and proposes exact and approximate network models that include the effect of the spectrum contiguity and spectrum continuity constraints, spectrum conversion, and defragmentation.

Spectrum fragmentation, if not handled well, can lower the spectrum efficiency in EONs, which leads to higher connection blocking. The smaller gaps formed between occupied spectrum (i.e., fragmentation) can be filled or utilized by splitting larger bandwidth

demands into smaller units (e.g., 100 Gb/s into ten 10 Gb/s substreams), and transmitted over multiple paths [11]. However, the traffic splitting and aggregation require managing multiple transponders and on route switches, which increases complexity and cost, in addition to spectrum wastage due to multiple guard bands [14]. Therefore, an obvious option is to utilize efficient spectrum allocation policies for connection admission in EONs without splitting the bandwidth demands [24, 25, 26, 27, 28, 29], which establishes new requests such that the spectrum fragmentation is minimal. Additionally, fragmentation is handled not only the spectral domain but also in time and spatial domains [25, 29], since connections usually have different holding (service) times, and they traverse on different paths in the network. A detailed survey of various RSA schemes can be found in [30], and the fragmentation issue is covered in a subsequent article [31]. However, as network traffic varies with time, defragmentation as a connection reconfiguration scheme is desirable, since spectrum allocation might not be optimal. The defragmentation in general is an NP-hard problem in an EON [32]. In the literature, both proactive and reactive defragmentation schemes have been proposed to minimize the fragmentation in spectral domain in EONs. Proactive methods [33, 34, 35, 36, 37], are carried out either periodically or when setting up a new connection by using performance indicators like *fragmentation index* (FI) as a measure of spectrum fragmentation or using resource partitioning method [38, 39]. Reactive methods [33, 34, 37, 40, 41, 42, 43, 44], on the other hand, reconfigure some or all existing connections when a connection request can not be established for the reasons that the spectrum is fragmented and not because it is insufficient to accommodate the connection request.

Although the connection reconfiguration as a defragmentation method lowers the connection blocking probability in EONs and increases spectrum utilization, it is generally undesirable as it usually

results in some form of service disruption. To this end, studies have been carried out to minimizing disruptions during reconfigurations, or their complete elimination [34, 45, 46, 47]. To avoid service disruptions, a connection can be moved from one frequency f_0 to another f_3 if the intermediate spectrum (i.e., from f_1 to f_2) on the same route is free. In this way, spectrum fragmentation can be minimized in the so-called *non-disruptive* fashion. However, non-disruptive reconfiguration has its own limits in the extent to which it can minimize the blocking in the network; clearly reconfiguration of the connections can bring more benefits. Hence, the question is whether other methods can increase the resource utilization while minimizing the service disruptions, which is also addressed in this thesis.

We further envision that with the time-varying nature of traffic in fiber-based access networks, wireless networks and cloud data centers networks, for which EONs serve as core networks, they would require intelligent resource management techniques, and static planning of resources is not enough to meet the applications' requirements in terms of bandwidth, latency, deadline, etc. Flexi-grid technologies are likely to be deployed in access networks as well as optical data center networks (ODCNs) to cope up with the increase in traffic and also to mitigate the overall network power consumption [48]. The good thing with the traffic is that in most of the scenarios, e.g., access networks, data center networks, traffic volume carried over the networks exhibit varying trends over the day, and it could be predicted and classified using machine-learning methods [49]. Thus, given the traffic forecast and acknowledging its periodicity, there is a need for applying learning algorithms for better resource utilization in EONs in general and datacenter networks in particular.

This thesis addresses three core challenges that a network resource management system should address in EONs. The first part models the RSA schemes in EONs through exact and approximate methods with the consideration of the spectrum contiguity and spectrum con-

tinuity constraints, spectrum conversion, and spectrum defragmentation. Notice that approximate models are proposed for analyzing the blocking probability in large scale EONs, since exact models are tractable only for very small networks. The results show that connection blocking in EONs happens mainly due to fragmentation of occupied optical spectrum, and it can be reduced significantly up to moderate loads using defragmentation schemes or deploying spectrum converters on the network switches, however, it is unavoidable in a mesh network. The analytical modeling in Chapters 2 and 3 is performed for two well-known spectrum allocation policies, namely random-fit and first-fit, which are simple, yet complex to model, and do not consider fragmentation factor for connection establishment. Therefore, Chapter 4 proposes an efficient resource allocation algorithm in EONs in addition to proposing a proactive and a reactive resource reallocation schemes to increase spectrum utilization and reduce connection blocking probability in EONs. Finally, in Chapter 5 this thesis addresses the key question: whether machine learning-based algorithms are the solutions for efficient management of network resources? To this end, it presents two use-cases of machine learning for resource provisioning in optical data center networks, and fiber-wireless networks.

1.1. Thesis Contributions

The work in this thesis contributes to analytical modeling and algorithms related to resource (re)allocation schemes, and presents three use cases of machine learning in managing optical network resources with a focus on the efficient utilization of resources and securing EONs. The primary contributions of this thesis is to model routing and spectrum (re)allocation schemes, namely a κ-shortest path routing and a first-fit or a random-fit spectrum allocation scheme, in EONs with both exact and approximate Continuous Time Markov Chain (CTMC)-based approaches. As analytical modeling is complex for adaptive resource provisioning schemes, heuristic approaches for multi-dimensional resource (re)allocation schemes are proposed in this thesis to provision spectrum resources efficiently in EONs. Furthermore, emerging applications require not only the efficient provisioning of resources but also an intelligent management of the provisioned resources throughout their service periods. Thus, with the help of machine learning techniques, the thesis also addresses the challenges related to time-varying resource (re)allocation in optical datacenters as well as in fiber-wireless networks with a special usecase of animal human cohabitation. The overall contribution is divided into three major categories: stochastic analysis, algorithms, and application of machine learning, which are briefly summarized next.

1.1.1. Analysis of Resource Allocation in Elastic Optical Networks

The network dimensioning is an important problem in networking, which deals with the problem of allocating capacity to the network links in order to meet the demands of users. Recognizing the fact that spectrum fragmentation is inevitable in mesh-topology-based EONs, this thesis proposes the first ever exact modeling of the RSA

problem in EONs using CTMC approaches. The modeling of the RSA problem is fundamental to computing the connection blocking probability in any optical networks. What makes the RSA problem complex to model is the two constraints: spectrum contiguity and continuity, in addition to the fragmentation created by non-uniform bandwidth in a mesh network. These two constraints along with the resulting spectrum fragmentation make the RSA a NP-hard problem, for which the exact blocking model has been formulated for the first time in this thesis. We address this grand challenge by proposing an exact Markov model for computing connection blocking in EONs, where a network state is represented by occupancy of individual spectrum slices on all network links. Notably, we analyze a κ-shortest path routing, also called as a fixed alternate routing in Chapter 2, that selects a first admissible path out of κ_o shortest paths between a node-pair. For the spectrum allocation part, we analyze two popular schemes: Random Fit (RF), which randomly allocates spectrum to connection requests; and First Fit (FF), which allocates first available slices to a new bandwidth request. Since the complexity of the exact Markov models grows exponentially with the link capacity, number of links, routes, and classes of demands, we further advance the state-of-the-art in computing approximate blocking probability in EONs using an independence link assumption. We develop load-independent and load-dependent approximations and derived unique formulas to compute probability of acceptance of requests in a given occupancy state. This thesis compares the novel approximate blocking performance with the exact and/or simulation results to evaluate the approximation analysis. The approximate results are accurate under a range of scenarios, including varying link capacities, classes of demands, traffic loads and network topology.

1.1.2. Analysis of Resource Reallocation in Elastic Optical Networks

As we argued previously that connections need to be reconfigured proactively or reactively to minimize connection blocking in EONs, this thesis also proposes an analytical framework to model a defragmentation process in a generalized form that allows a proactive, a reactive and a combined triggering mechanism for connection reconfiguration in a link using a CTMC model. Furthermore, we propose an exact network-wise analysis of the impact of a reactive defragmentation process on blocking due to resource unavailability, called as resource blocking, and due to spectrum fragmentation, referred as fragmentation blocking, separately in EONs for the first time. The exact defragmentation model considers occupancy status of spectrum slices of all network links and defines additional defragmentation states using the number of serving and waiting connections in order to compute blocking in a defragmentation-enabled EON. For the scalability reason, we utilize the proposed load-independent and load-dependent approximation models for EONs without defragmentation together with the independence link assumption in order to obtain approximate resource and fragmentation blocking probabilities in large scale networks. The analytical (exact and approximate) results presented in this thesis are very interesting as they find a fraction of overall blocking, i.e., fragmentation blocking, that can not be eliminated in a mesh topology under two given RSA policies, the random-fit and the first-fit, even with an optimal defragmentation scheme.

1.1.3. Efficient Resource (Re)allocation Schemes in Elastic Optical Networks

The modeling of resource (re)allocation schemes is often limited in scope due to their underlining assumptions. Moreover, we observe

that despite multiple dimensions (spectral, time, spatial) for resource allocation, EONs are prone to spectrum fragmentation. The spectrum fragmentation issue can be addressed either in the connection admission phase, or during a spectrum defragmentation phase via reconfigurations of existing connections. Although an optimal RSA solution can be obtained for a given traffic matrix in a very small scale EON, heuristic approaches are needed for real (large) networks even though they provide sub-optimal solutions. Towards a heuristic solution to the RSA problem, this thesis proposes an efficient resource allocation algorithm taking into account the multi-dimensional occupancy status of spectrum resources to minimize the connection blocking probability while maximizing the spectrum utilization. The proposed algorithm considers fragmentation in three dimensions – spectral, time and spatial, and develops a combined fragmentation metric to minimize fragmentation while setting up a new lightpath for admitting connection requests.

To address the issue of spectrum fragmentation in a dynamic environment, this thesis proposes a proactive and a reactive defragmentation schemes as methods to reallocate spectrum to some or all of existing connections such that fragmentation remains at a minimum level in EONs. The proposed proactive defragmentation method reconfigures some of the connections when admitting a new connection. This method takes into account the remaining lifetimes of existing connections and reallocates spectrum to them such that they do not cross-over, which is essential to provide non-disruptive reconfigurations. Additionally, we propose a reactive defragmentation approach, which is triggered when a new connection request can not be admitted without connection reconfigurations and the network fragmentation crosses a well-defined fragmentation threshold. The reactive scheme although provides lower blocking, it is disruptive in nature, since a graph-theoretic approach is utilized to relocate some of existing connections in order to find a required continuous

and contiguous free slices as required by a new connection request. We recommend that the reactive defragmentation scheme should be applied on top of a simple connection admission method like the first-fit, instead of using a complex connection admission scheme.

1.1.4. Application of Machine Learning in Resource Provisioning

Among many artificial intelligence-based technologies, machine learning is the most preferable technique for learning, classification/identification and prediction. This thesis presents two application scenarios related to resource (re)allocation: one in optical datacenter networks, and other in fiber-wireless networks, where machine learning algorithms, such as recurrent neural networks, convolutional neural networks, support vector machines, etc. can be used for prediction, classification activities, as well as for decision making. In the resource (re)allocation in optical datacenters, where holding (service) time information of applications is not known in advance and are heavy-tailed distributed, we predict lognormally-distributed mean holding time information of traffic classes, and estimate residual lifetimes of connections to (re)allocate spectrum resources efficiently. Finally, this thesis presents the resource allocation in fiber-wireless networks, with the aim to build an early warning system framework. The network handles different classes of traffic demands (animal, cellular and fiber-to-the-home), and allocates optical, wireless and computing (energy) resources intelligently. Precisely, various traffic classes are predicted to decide whether to classify monitored animal data at sensors or at base stations, and when to transmit processed or unprocessed data to base stations in order to generate warning alert signals, if necessary, and to alert passing vehicles or humans. Additionally, we utilized OFDM-based passive optical networks to efficiently mange the upstream bandwidth of optical network units.

1.2. Supporting Publications

1.2.1. Journal Articles

- S. K. Singh and A. Jukan, "Computing Exact and Approximate Blocking Probabilities in Elastic Optical Networks," arXiv preprint arXiv:1706.07747. Available online: https://arxiv.org/pdf/1706.07747.pdf.

- S. K. Singh, F. Carpio, and A. Jukan, "Improving Animal-Human Cohabitation with Machine Learning in Fiber-Wireless Networks," *MDPI Journal of Sensor and Actuator Networks*, Special Issue on Optical Technologies for IoT, Smart Industry and Smart Transportation, vol. 7, no. 3, 2018. https://doi.org/10.3390/jsan7030035.

- S. K. Singh and A. Jukan, "Machine Learning-based Prediction for Resource (Re)allocation in Optical Data Center Networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, issue 10, pp. D12-D28, 2018. https://doi.org/10.1364/JOCN.10.000D12.

- J. Mata, I. de Miguel, R. J Durán, and N. Merayo, S. K Singh, A. Jukan, and M. Chamania, "Artificial intelligence (AI) methods in optical networks: A comprehensive survey," *Optical Switching and Networking*, vol. 28, pp.43-57, 2018. https://doi.org/10.1016/j.osn.2017.12.006

- S. K. Singh and A. Jukan, "Efficient Spectrum Defragmentation with Holding Time Awareness in Elastic Optical Networks," *IEEE/OSA Journal of Optical Communications and Networking,* vol. 9, issue 3, pp. B78-B89, 2017. https://doi.org/10.1364/JOCN.9.000B78.

- S. K. Singh, W. Bziuk and A. Jukan, "Analytical Performance Modeling of Spectrum Defragmentation in Elastic Optical Link

Networks," *Optical Switching and Networking,* vol. 24, pp 25-38, 2016. https://doi.org/10.1016/j.osn.2016.11.001.

- S. K. Singh, T. Das, A. Jukan, "A Survey on Internet Multipath Routing and Provisioning," *IEEE Communications Surveys and Tutorials,* vol. 17, no.4, pp 2157-2175, 2015. https://doi.org/10.1109/COMST.2015.2460222.

1.2.2. Conferences

- S. K. Singh and A. Jukan, "Computing Blocking Probabilities in Elastic Optical Networks with Spectrum Defragmentation," *IEEE International conference on Computer Communications (INFOCOM)*, Paris, April 2019.

- J. Mata, I. de Miguel, R. J Durán, and N. Merayo, S. K Singh, A. Jukan, and M. Chamania, "Application of Artificial intelligence Techniques in Optical Networks," *IEEE Photonics Society Summer Topical Meeting Series (SUM)*, Hawaii, USA, July 2018.

- S. K. Singh, W. Bziuk and A. Jukan, "A Combined Optical Spectrum Scrambling and Defragmentation in Multi-Core Fiber Networks," *2017 IEEE International Conference on Communications (ICC),* Paris, France, May 2017.

- S. K. Singh and A. Jukan, "Holding-Time Information (HTI): When to Use it?," in *2017 Optical Fiber Communication (OFC) Conference,* Los Angeles, California, USA, March 2017.

- S. K. Singh, W. Bziuk and A. Jukan, "Balancing Data Security and Blocking Performance with Spectrum Randomization in Optical Networks," in *IEEE Global Communications Conference (GLOBECOM),* Washington, USA, December 2016.

- S. K. Singh and A. Jukan, "Non-Disruptive Spectrum Defragmentation with Holding-Time Awareness in Optical Networks," in *Optical Network Design and Modeling (ONDM),* Spain, May 2016.

- S. K. Singh, W. Bziuk and A. Jukan, "Balancing Security and Blocking Performance with Reconfiguration of the Elastic Optical Spectrum," in *International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO),* Croatia, May 2016.

- S. K. Singh, W. Bziuk and A. Jukan, "Defragmentation-as-a-Service (DaaS): How beneficial is it?," in *Optical Fiber Communication (OFC) Conference,* Anaheim, CA, USA, March 2016.

1.3. Thesis Organization

This thesis is structured in five chapters. After this introductory chapter, Chapter 2 explains the fragmentation issue in EONs, and presents the theoretical analysis of the κ-shortest path routing and two different spectrum allocation schemes: RF and FF to computing the exact and approximate blocking probabilities in EONs. Chapter 3 analyses the effect of defragmentation in reducing connection blocking probability in EONs. The fourth chapter proposes algorithmic approaches to efficiently (re)allocating optical resources with the aim of maximizing spectral utilization in EONs. Chapter 5 provides two application scenarios of machine learning techniques to managing optical resources in optical datacenter networks and fiber–wireless networks. Finally, the sixth chapter concludes the thesis.

2

Analysis of Resource Allocation in Elastic Optical Networks

2.1. Introduction

Elastic optical networks (EONs) present many advantages, such as increasing spectrum utilization, reducing connection blocking probability, supporting multi-rate client signals (from Gigabit/s to Terabit/s) by means of multiplexing several OFDM or Nyquist subcarriers into a single entity called a super-channel, enabling optical network virtualization, etc. over traditional WDM networks by allocating *just enough* spectrum to heterogeneous bandwidth demands. The flexible spectrum allocation and multi-rate support of client data rates can improve spectrum utilization up to 95% compared to fix-grid WDM networks [50]. At the same time, flexible optical transponders and switching technologies also raise new challenges for future optical networks. One such challenge is the requirement in establishing an optical channel called lightpath between an origin-destination (OD) pair using a single transponder per node in an EON, wherein intermediate switching nodes do not have spectrum conversion capabilities (which is analogous to wavelength conversion). Precisely, each lighpath must be established over adjacent slices, called spectrum contiguity constraint, and use same

spectrum slices on all traversing links of its routing path, called as spectrum continuity constraint. Furthermore, as network traffic varies over time, spectrum gets fragmented due to the setting up and tearing down of non-uniform bandwidth requests under these two constraints, which leads to a non-optimal spectrum allocation, and hence results in a relatively higher blocking probability and lower spectrum utilization in EONs. Therefore blocking analysis in EONs is analogous to the modeling of the routing and spectrum allocation (RSA) schemes while taking into the effect of the *spectrum contiguity* and *spectrum continuity* constraints along with the resulting spectrum fragmentation, which has been prime focus of researchers since the evolution of EON technologies in the last decade, i.e., 2010 onward. Although there have been some progress towards exact and approximation approaches, modeling the RSA schemes in either exact or approximate way is shown not to be a trivial task either [21, 22, 51] due to need to consider both the RSA constraints and spectrum fragmentation factor.

This chapter first explains the fragmentation issue in EONs and related RSA models. The following section presents the first *exact Markov model* for an EON to obtain the connection blocking under the κ-shortest path routing and a random-fit (RF) or a first-fit (FF) spectrum allocation schemes. Following which, due to scalability issue of the exact network model, we propose a *reduced state Markov model* to compute approximate blocking probabilities in EONs, with and without spectrum conversion, using various load-independent and load-dependent approximations.

2.2. Supporting Publications

- S. K. Singh and A. Jukan, "Computing Exact and Approximate Blocking Probabilities in Elastic Optical Networks," arXiv preprint arXiv:1706.07747. Available online: `https:`

//arxiv.org/pdf/1706.07747.pdf.

- S. K. Singh and A. Jukan, "Computing Blocking Probabilities in Elastic Optical Networks with Spectrum Defragmentation," *IEEE International conference on Computer Communications (INFOCOM)*, Paris, April 2019.

2.3. Fragmentation Issue and RSA Models

In EONs spectrum fragmentation occurs mainly due the spectrum continuity and contiguity constraints, and connection setup and tear-down of heterogeneous demands in a dynamic environment under a given RSA policy. Fig. 2.1 shows how a connection between a node-pair A and E with a bandwidth demand of $d_1 = 1$ slice is setup over the same (continuous) slice s_1 on each link of its path $A-B-D-E$. Another connection between node-pair C and D is shown to be setup over a slice s_2 (either due to a random allocation policy or a connection between $C-D$ over a slice s_1 might have departed), which separates the two remaining free slices (s_1 and s_3) on the link $C-D$. Let us now assume that a new connection request over a path $C-D-E$ arrives with a demand of $d_2 = 2$ consecutive slices. Then, it is easy to see that even though both links $C-D$ and $D-E$ have two free slices, the free slices s_1 and s_3 are neither contiguous (adjacent) on the link $C-D$ nor aligned (continuous) to both free slices s_2 and s_3 of the link $D-E$. Notice that, had the connection between $C-D$ been established over a slice s_1, or if it is reconfigured to slice s_1 then the connection request between C and E could have been accepted over the slices s_2 and s_3. Thus, the spectrum fragmentation increases connection blocking probability and reduces spectrum utilization in EONs.

The analytical modeling of the RSA problem is similar to the modeling of the RWA problem in WDM networks with an added constraint of spectrum contiguity, which makes the RSA modeling

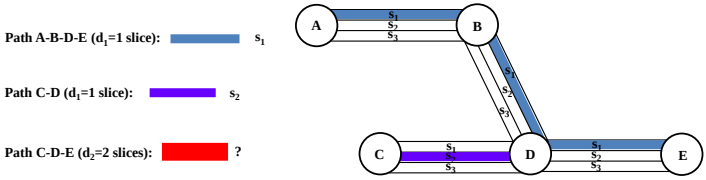

Figure 2.1.: An example of spectrum allocation with the continuity and contiguity constraints, and resulting spectrum fragmentation in an EON with fiber links with capacity $C = 3$ slices.

a challenging task. Most of the analytical work previously reported, starts with an analysis of connection blocking using CTMC models in a single optical link, – which is due to complexity, building it up to a computation of blocking in a network. In the CTMC-based models, network states – formed by allocation and deallocation of spectrum to arriving and departing connection requests, respectively – and transition rates between states are two important things. The computation of these two entities defines the model accuracy. For example, when the network states and transition rates are defined based on the exact occupancy of all links, and arrival and departure rates, then the network can be modeled in an exact manner. In [20] and [52], an exact blocking probability of a single link was analyzed by modeling the bandwidth occupancy using a CTMC model under the RF and FF policies. A so-called exact blocking analysis in EONs was given by Beyranvand *et al.* [22], where network states are defined by a spectrum union operation of exact link-states on a multi-hop route. Defining a network state by the spectrum union operation of exact link-states does not differentiate among spectrum patterns (exact state) formed by overlapping routes, and the transition rates are thus approximately computed, as the load correlation factor among links is ignored. Therefore, the presented exact results

for a 2-hop path in [22] do not match simulation results.

Because the complexity of an exact link or network model grows exponentially with link capacity or network size, there have been some progress towards approximation approaches, which is shown not to be a trivial task either [21, 22, 51, 53]. In an approximate approach, a state can be defined by, for example, the total occupied slices per link or the number of connections per class. However, it is analytically hard and computationally challenging to taking into account the effect of fragmentation in an approximate model while deriving the connection setup rate, i.e., the effective arrival rate at which a link allows connections to be setup in a given spectrum occupancy state. It is hard because a given number of occupied slices could be represented by fragmented as well as non-fragmented spectrum patterns, and the fragmented states could not accept an incoming connection request. Moreover, the non-polynomial complexity makes its computation intractable [54,55]. Therefore, in [21], authors used a binomial distribution to compute the probability of a required free continuous and consecutive slices on a route using the average carried load per slice as a slice occupancy probability. The same approach was used in [22] to approximate an idle slice probability using a link-state probability distribution, which is obtained by a Kaufman's formula [56] without considering valid spectrum patterns (exact states) and the RSA constraints. More in detail, the Kaufman blocking probability solutions in [22] were shown to match the exact analysis, and simulation results, for cases where RSA constraints are relaxed. The binomial approach [21, 22] was shown useful for a small scale link and networks. [53] proposed an iterative procedure – a parallel birth-death process having different initial states, to approximate path request (connection) blocking in EONs enabled with spectrum converters. Note that all above mentioned works consider a fixed routing scheme, where a connection request between a node-pair is always routed over a fixed path, even though alternate paths

might be existing, which often decreases spectrum utilization and increases connection blocking probability in optical networks [57].

To advance the previous studies, the next section introduces an *exact Markov model* for an EON, wherein occupancy of each slice on all links is considered to obtain the connection blocking probability under the κ-shortest path routing and the RF or FF spectrum allocation schemes. Furthermore, due to scalability issue of the exact network model, we propose a *reduced state Markov model*, wherein link-states are represented by total occupied slices, to compute approximate blocking probabilities in EONs, with and without spectrum conversion, using load-independent and load-dependent approximations. Additionally, we consider a *reduced-load approximation*, which helps in calculating the effective load of the combined connections on a link, and an *independence link* assumption, i.e., spectrum occupancies of links are statistically independent, using a multirate loss model. We note that the multirate loss model [54, 55] – originally developed for fix-grid Wavelength Division Multiplexed (WDM) networks – uses request acceptance probability terms to compute connection setup rates and blocking probability. However, without using the exact network model, the exact calculation of the acceptance probability in an EON, which also needs to consider contiguity constraint, is still an unsolved and computationally challenging problem, as it generally requires a non-polynomial computation. Thus, we propose three different load-independent approximations: (i) Equiprobable Occupancy Vector (EOV), (ii) Uniform, and (iii) Equiprobable Exact States (EES). All these approximations work on an equiprobable assumption that considers different sample spaces. A sample space is a set of entities, e.g., all exact link occupancy states on a route, all spectrum slices, or only exact link-states having the same number of occupied slices, which are assumed to be equiprobable (explained in Section 2.5.2). Notice that these approximations do not handle slice occupancy correlation (SOC) among

Table 2.1.: Comparison of related and this work w.r.t. various factors used in BP computation in EONs without SC [58].

BP	work	Approach	Routing	Exact states	Contiguity	Continuity	RL App.	Correlation
Exact	[22]	CTMC	Fixed	Partially	✓	✓	-	✗
	Thesis	CTMC	FAR	✓	✓	✓	-	✓
App.	[22]	Kaufman [56]	Fixed	✗	✗	✗	✗	✗
		Binomial [21]	Fixed	✗	Partially	Partially	✗	✗
	Thesis	EES [55]	FAR	Partially	✓	Partially	✓	✗
		SOC [55]	FAR	Partially	✓	Partially	✓	Partially
		EOV [55]	FAR	✓	✓	✓	✓	✗
		Uniform [55]	FAR	✗	Partially	Partially	✓	✗

App.: Approximation; RL: Reduced-load; FAR: Fixed-Alternate Routing; EES: Equiprobable Exact States; SOC: Slice Occupancy Correlation; EOV: Equiprobable Occupancy Vector.

fragmented and non-fragmented exact states differently, which is important especially for low network loads and the FF scenarios. Therefore, we propose a load-dependent SOC approximation to calculate the probability of acceptance of a request in EONs considering the fragmentation and average link occupancy for a given network load. Moreover, the EES and SOC approximations for computing the request acceptance probability on a route are given as a product-form to reduce the complexity to a polynomial time.

To illustrate how the proposed models advance the state-of-the-art, Table 2.1 summaries the main factors of our analysis and the related work in computing exact and approximate blocking probabilities (BPs) in EONs, all without spectrum conversion (SC). (SC means that if, for instance, two contiguous slices s_1 and s_2 are used on link 1, they can be *converted* in an intermediate optical node and allocated to contiguous slices s_3 and s_4 on subsequent link 2. A comparison of different approaches for BP computation in EONs with SC can be given by omitting the continuity constraint.). Unlike the fixed (or shortest path) routing used in the related works, we model a Fixed routing ($\kappa_o = 1$), as well as a Fixed Alternate Routing, FAR ($\kappa_o \geqslant 2$). In the FAR, a connection is attempted to route sequentially over an ordered set of κ_o shortest paths until an admissible path is found out of κ_o available shortest paths. Obviously, approximation approaches, including ours, do not consider

exact network states (i.e., valid spectrum patterns). However, the EES and SOC (EOV) approaches consider exact link-states for the RF (given RSA) scenario. The contiguity and continuity constraints are shown only partially true for a method that does not consider valid spectrum patterns while computing the availability of a free block of contiguous and continuous slices on a route. Furthermore, all approximations assume link independence, and only the SOC approach considers load correlation on slices of a link. Notably, we compute exact and approximate blocking probabilities for a given number of κ_o paths per node-pair under four operation modes: RF policy without SC (RF), FF policy without SC (FF), RF policy with SC (RF-SC), and FF policy with SC (FF-SC), where all operation modes assume that the spectrum contiguity constraint must be satisfied while admitting a request. Also notice that the scenarios with SC make sense only in multi-hop routes in EONs, as SC helps in relaxing the continuity constraint.

2.4. Exact Blocking Analysis in EONs

The RSA problem of can be modeled by a CTMC approach [59], where a stochastic process $\{X(t), t \geq 0\}$ denotes a network state (say V_i) at time t, and takes values on a finite or countable network state space, say, $\Omega_V = \{V_1, V_2, \ldots, V_i, \ldots, V_{|\Omega_V|}\}$ with the following Markov or memoryless property: the future, $\{X(s+t) : t \geq 0\}$, given the present state, $X(s)$, is independent of the past, $\{X(u) : 0 \leq u < s\}$, i.e.,

$$P(X(s+t) = V_j | X(s) = V_i, \{X(u) : 0 \leq u < s\}) \\ = P(X(s+t) = V_j | X(s) = V_i) = P_{ij}(t) \quad (2.1)$$

where $P_{ij}(t)$ is the probability that the network will be in a state V_j, t time units from now, given it is in state V_i now. In the continuous time case, the intensity or transition rate from the state V_i to

$V_j, j \neq i$, q_{ij} is defined as $\lim_{t\to 0^+} \frac{P_{ij}(t)}{t}$. The Poisson arrival process, which exhibits exponential interarrival time distribution, is a simplest Markovian arrival process with memoryless property, which helps to yield analytical results. For this reason, most of the previous work, including this thesis, consider the exponential connection interarrival with rate λ and exponential holding time distribution with rate μ. Therefore, for the exponential distribution, the intensity $q_{ij}, j \neq i$ at which the transition from a state V_i to another state V_j could occur due to an arrival (departure) event is same as its mean arrival (departure) rate λ (μ) [59]. It is important to mention that the assumptions of Poisson arrival and exponential holding times are generally not true for most of the real world applications, but they are useful in providing insights on the proficiencies of various algorithms that could use true arrival and departure processes of connections in optical networks.

To compute the exact blocking probability in EONs, we present an exact CTMC model that models a κ-shortest path routing and RF and FF spectrum allocation policies. The notations and definitions of some of the parameters used in the model are listed in Table 2.2. All assumptions for the CTMC model are listed below for an arbitrary EON topology with N nodes, J unidirectional fiber links (belongs to set $\mathcal{J}$), and C spectrum slices per link.

- Arrivals of class $k \in \{1, 2, \cdots, K\}$ connection path requests between an origin-destination (OD) node-pair $o(\in \mathcal{O})$, in short (o, k), follow Poisson process with arrival rate λ_k^o , and connection holding (service) time is exponentially distributed with mean $1/\mu_k$. We assume that the arrivals and departures are statistically independent.

- Each OD pair request $o \in \mathcal{O}$ is attempted to be routed on a path $r(o, m) \in R(o)$ sequentially in a set of κ_o link-disjoint shortest paths belonging to $R(o) = \{r(o, m), m = 1, 2, \ldots, \kappa_o\}$.

- Spectrum is allocated according to a policy (e.g., RF) on the first route $r(o,m)$ that has sufficient ($\geqslant d_k$) contiguous and continuous free slices. However, when network nodes are equipped with spectrum converters (e.g., RF-SC), then the continuity constraint is omitted.

Table 2.2.: Notations and the parameters used in the models [58].

Notation	Description
C	Total number of spectrum slices (or capacity units) per link
K	Number of connection classes. Note that classes $k = 1, 2, \cdots, K$
λ_k^o	Arrival rate of class k connections of an OD pair $o \in \mathcal{O}$
μ_k	Service rate of a class k connection; mean service time $t_c = 1/\mu_k$
$\boldsymbol{d}$	$\equiv (d_1, d_2, \ldots, d_K)$, where d_k: bandwidth (in slices) of class k
S	$\equiv (s_1, s_2, \ldots, s_C)$, where s_c: free or occupied state of an c^{th} slice
$\boldsymbol{n}$	$\equiv (\boldsymbol{n}^1, \boldsymbol{n}^2, \ldots, \boldsymbol{n}^{\mathrm{r}})$, where $\boldsymbol{n}^o$ is the set of connections $\in o$.
$\boldsymbol{n}^o$	$\equiv (n_1^o, n_2^o, \ldots, n_K^o)$, where n_k^o is the # of class k connections $\in o$.
$n_k^o(\boldsymbol{n})$	Number of class k connections of an OD pair o in $\boldsymbol{n}$
$n_k^o(V_i)$	Number of class k connections of an OD pair o in a network state V_i
$\Gamma_{V_i}^{o,k+}$	Set of possible states after a class k request arrives on OD pair o in V_i
$\Gamma_{V_i}^{o,k-}$	Set of possible states after a class k request on OD pair o departs from V_i
$f_l(S_i)$	Size of the largest block of consecutive free slices in a link-state S_i
$\lvert\Omega_S(x)\rvert$	# of all exact link-states $S_i \in \Omega_S$ representing total occupancy of x slices
$\lvert\mathbb{NB}(x,k)\rvert$	# of non-blocking exact states with occupancy x for class k requests
$\lvert\mathbb{FI}(x,k)\rvert$	# of fragmented exact states with occupancy x for class k requests

2.4.1. Generation of Exact States, and State Transitions

Network states are created by the (de)allocation of spectrum to various classes of connections between different OD node-pairs $o \in \mathcal{O}$, and they differ with respect to the occupancy status of slices. We define a network state by a $|\mathcal{J}| \times C$ matrix, where an element (j, c)

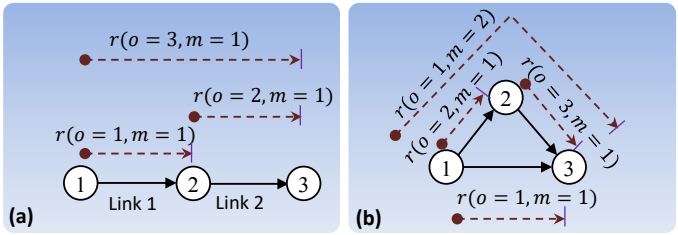

Figure 2.2.: (a) A 2-link topology with 3 OD routes. (b) A 3-node network with 3 OD pairs and 4 routes.

represents the free or occupied status of a c^{th} slice on a link j, $j = 1, \ldots, |\mathcal{J}|; c = 1, \ldots, C$. Let us represent a free slice by 0, and an occupied slice by either r' or ∞ depending on whether the occupied slice is the start or the remaining bandwidth occupancy of a class k connection on an m^{th} route of an OD pair o, i.e., $r(o, m)$, which can be assigned a route number as $r' = (o-1)\kappa_o + m$. For example, the assignment of route numbers is shown for a 2-link and a 3-link networks in Fig. 2.2.

Furthermore, Fig. 2.3 illustrates the exact states and transition rates related to an empty network state V_1 in a 2-link EON [shown in Fig. 2.2(a)] with 2 slices fiber-link (i.e., $C = 2$) under the RF scenario without SC. Here, a network state V_1 without any connection is represented by $V_1 = (S_1^1, S_1^2) = ((s_1^1, s_2^1), (s_1^2, s_2^2)) = ((0, 0), (0, 0))$, where S_i^j represents the i^{th} network state occupancy on a link $j, j = 1, 2$, and s_c^j shows the free or occupied status of an ordered (left to right) slice $c, c = 1, 2$ on a link j. Now, to illustrate the formation of a few other states, let us assume that a new class 1 connection request belonging to an OD pair $o = 1$ arrives in an empty state V_1 with a bandwidth demand $d_1 = 1$ slice in the same 2-link network. Since there is only a single route per OD pair o, i.e., $m = \kappa_o = 1$, the spectrum can be allocated in one of two different

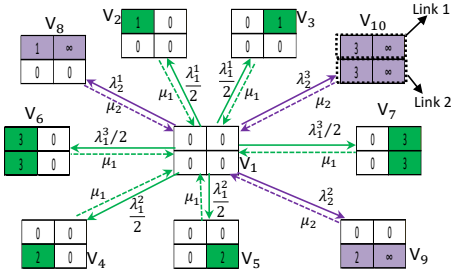

Figure 2.3.: State transitions in a 2-link topology from and into a network state V_1 in a 2-link network with link capacity $C = 2$ slices are shown due to arrivals and departures of 3 OD routes with demands $\boldsymbol{d}$=(1,2) slices under the RF scenario without SC.

ways under the RF scenario, as shown by network states V_2 and V_3 in Fig. 2.3, where the top and bottom link-states represent spectrum occupancy on links 1 and 2, respectively. Similarly, an arrival of a class 2 request with demand $d_2 = 2$ consecutive slices on a route $r(o = 2, m = 1)$ or $r(o = 3, m = 1)$ in the state V_1 will cause the network state to transit to a state V_9 or V_{10}, respectively. In the FF scenario, on the other hand, an arrival ($d_1 = 1$ slice) on a route $r(o = 1, m = 1)$ in an empty state V_1 will trigger the network transition to only one network state $V_2 = ((1, 0), (0, 0))$, where the first slice is allocated on link 1 in Fig. 2.4. The RF-SC (FF-SC) tries to first allocate a request over a random (first) set of contiguous and continuous free slices, and the continuity constraint is relaxed when such spectrum paths are not found. Algorithm 1 describes a method to create valid exact states under a given scenario M:= RF, FF, RF-SC, FF-SC, defines transitions between states using functions $\Gamma^{o,k\pm}_{V_i}$, and identifies blocking states using $\mathbb{B}(\cdot)$, which is explained below.

Algorithm 1 initializes a network state in an empty state V_1, i.e.,

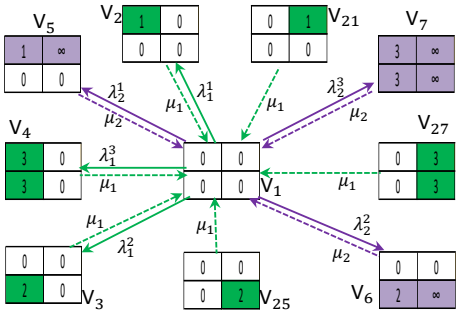

Figure 2.4.: State transitions in a 2-link topology from and into a network state V_1 in a 2-link network with $C = 2$ slices, and demands $\boldsymbol{d}$=(1,2) slices under the FF scenario without SC.

$s_c^j = 0, \forall j, c$. Next, in steps 4–10 for each combination of OD pairs and classes of demands, a state V_i tries to allocate d_k spectrum slices to a class k request on a first route $r(o,m)$, $m = 1$ to κ_o, which has d_k free slices on each associated link as well as satisfies the RSA constraint(s) under a given RSA scenario M (e.g., RF, FF, RF-SC, FF-SC). When the spectrum allocation is possible for an arrival of a connection request (o,k) in the state V_i, its possible target state(s) is (are) stored in a set $\Gamma_{V_i}^{o,k+}$ in step 6. For example, in Fig. 2.3, the set of target states of the state V_1 due to an arrival of a 1-slice demand on route 1 (or link 1) is $\Gamma_{V_1}^{1,1+} = \{V_2, V_3\}$. If the spectrum allocation is possible on a route $r(o,m)$ (i.e., $\Gamma_{V_i}^{o,k+} \neq \emptyset$) then only new states ($\notin \Omega_V$) are appended to Ω_V in step 8, and the remaining larger hop routes of o, i.e., $\{r(o,m+1), \ldots, r(o,\kappa_o)\}$ are not checked for spectrum allocation of the request (o,k). When no routes in $R(o)$ could allocate the required free spectrum to the request (o,k), then the state V_i is identified as a blocking state for the request (o,k)

Algorithm 1 Find network state-space Ω_V, transitions between states, and blocking states [58].

1: Given: RSA scenario M, , Network $(C, \mathcal{J}, \kappa_o, R(o), \forall o : \mathcal{O})$, demands $\boldsymbol{d} = (d_1, d_2, \ldots, d_K)$.
2: Initialize $\Omega_V \leftarrow V_1 := zeros_{|\mathcal{J}|\times C}$; $i \leftarrow 1$.
3: **repeat**
4: **for** $(o, k) = (1, 1) : (|\mathcal{O}|, K)$ **do** // spectrum allocation due to an arrival
5: **for** $m = 1 : \kappa_o$ **do** // check routes sequentially until it finds $\geqslant d_k$ free slices
6: $\Gamma_{V_i}^{o,k+} \leftarrow$States after possible allocation of d_k slices on a route $r(o, m)$ in the state V_i based on a given RSA scenario M.
7: **if** $\Gamma_{V_i}^{o,k+} \neq \emptyset$ **then**
8: For all $V_t \in \Gamma_{V_i}^{o,k+}$: if $V_t \notin \Omega_V \Rightarrow \Omega_V \leftarrow [\Omega_V, V_t]$; **Goto** step 10.
9: **end if**
10: **end for**
11: **if** $\Gamma_{V_i}^{o,k+} = \emptyset$ **then**
12: $\mathbb{B}(o, k) \leftarrow V_i$
13: **end if**
14: **for** $m = 1 : \kappa_o$ **do** // *due to Deallocation*
15: $\Gamma_{V_i}^{o,k-} \leftarrow$States after deallocation of spectrum of class k connection(s) from V_i on a route $r(o, m)$.
16: For all $V_t \in \Gamma_{V_i}^{o,k-}$, if $V_t \notin \Omega_V \Rightarrow \Omega_V \leftarrow [\Omega_V, V_t]$
17: **end for**
18: **end for**
19: $i \leftarrow i + 1$
20: **until** $i > |\Omega_V|$

in step 12. For example, the states V_2 and V_3 are blocking states for a class-2 (2-slices) but not for class-1 (1-slice) demands on the route $o = 1$ in Fig. 2.3, as they have only one free slice on route 1. New network states could also be created due to the departure of connections from all possible routes, mainly in the FF-related scenarios. Therefore, steps 14–17 try to capture new states and related transitions due to the spectrum deallocation process. For example, we can see in Fig. 2.4 that in the FF scenario the state V_1 can be reached from a state V_{21} due to a connection departure, but the transition due to a connection arrival in the state V_1 does not lead

to V_{21}. The algorithm converges when the (de)allocation processes are checked iteratively in all states in Ω_V. Note that the preference of smaller over larger hop paths generally helps in lowering blocking probability. However, the admission criteria can be easily extended to a fragmentation- or utilization-aware RSA by selecting a route with an additional constraint of least fragmentation or utilization [60].

The individual transition rate from a state V_i to another state V_t, i.e., $q_{it}, i \neq t$ is obtained by either arrival or departure of a connection (o,k) is given by Eq. (2.2), and $q_{ii} = -\sum_{t \neq i} q_{it}, i = 1, 2, \ldots, |\Omega_V|$.

$$q_{it} = \begin{cases} \frac{\lambda_k^o}{|\Gamma_{V_i}^{o,k+}|} & \text{if } V_t \in \Gamma_{V_i}^{o,k+}, \text{ for any } (o,k) \\ \mu_k & \text{if } V_t \in \Gamma_{V_i}^{o,k-}, \text{ for any } (o,k) \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

It should be noted that in the FF and FF-SC scenarios, the number of elements in a set $\Gamma_{V_i}^{o,k+}$, i.e., $|\Gamma_{V_i}^{o,k+}|$ is 1 if the allocation function $\Gamma_{V_i}^{o,k+} \neq \emptyset$ for any pair of (o,k), otherwise it is zero 0.

2.4.2. Exact State Probabilities and Blocking Analysis

After generating all exact states and transitions among them, the global balance equation (GBE) of a network state $V_i, i = 1, \cdots, |\Omega_V|$ can be obtained by

$$\left(\sum_{\substack{(o,k)=(1,1),\\ \Gamma_{V_i}^{o,k+} \neq \emptyset}}^{(|\mathcal{O}|,K)} \lambda_k^o \; + \sum_{(o,k)=(1,1)}^{(|\mathcal{O}|,K)} n_k^o(V_i)\mu_k \right) \pi(V_i) =$$

$$\sum_{t=1, t\neq i}^{|\Omega_V|} \left(\sum_{\substack{(o,k)=(1,1),\\ V_i \in \Gamma_{V_t}^{o,k+}}}^{(|\mathcal{O}|,K)} \lambda_k^o / |\Gamma_{V_t}^{o,k+}| \; + \sum_{\substack{(o,k)=(1,1),\\ V_i \in \Gamma_{V_t}^{o,k-}}}^{(|\mathcal{O}|,K)} \mu_k \right) \pi(V_t) \tag{2.3}$$

where, the left hand side (LHS) represents an output flow rate from the state V_i, having steady state probability $\pi(V_i)$, while the right hand side (RHS) represents an input flow rate into the state V_i. More precisely, the first (second) term in the LHS represents the output rate due to arrivals (departures) in (from) V_i, and the first (second) term in the RHS is due to arrivals (departures) in (from) other states V_t that lead to the state V_i. As an example, under the RF scenario, the GBE for a state V_1 in Fig. 2.3 is given by $(\sum_{o=1,k=1}^{o=3,k=2} \lambda_k^o)\pi(V_1) = \mu_1(\sum_{t=2}^{7} \pi(V_t)) + \mu_2(\sum_{t=8}^{10} \pi(V_t))$. Notice that transitions from and into the state V_1 occur due to arrivals in V_1 and departures from other states, respectively. However, for example, the GBE of a state V_2 would also include rate $\lambda_1^1/\Gamma_{V_1}^{1,1+} = \lambda_1^1/2$ (and μ_1) in its RHS (in LHS) due to an arrival (departure) of class 1 connection on OD pair $o = 1$ in (from) the state V_1 (V_2). Remember that in the FF and FF-SC, the number of elements in a non-empty set $\Gamma_{V_i}^{o,k+}$ is 1. Therefore, the GBE of a state V_1 under the FF scenario in Fig. 2.4 can be obtained as $(\sum_{o=1,k=1}^{o=3,k=2} \lambda_k^o)\pi(V_1) = \mu_1(\sum_{t=2}^{4} \pi(V_t) + \pi(V_{21}) + \pi(V_{25}) + \pi(V_{27})) + \mu_2(\sum_{t=5}^{7} \pi(V_t))$, which allocates only first set of free contiguous and continuous slices. Under the RF-SC and FF-SC scenarios, we can also write the GBE of a state, for example, V_1 using Eq. (2.3) by including additional transition rates in the RHS of above given GBE equations for V_1 under the RF and the FF, respectively, due to departures of a class 1 (1 slice bandwidth) connection on a route for $o = 3$ from two additional network states $((3,0),(0,3))$ and $((0,3),(3,0))$, which are exclusively created because of spectrum conversion at node 2 in Fig. 2.2(a).

Under the stationary condition, the network state probabilities $\boldsymbol{\pi} = [\pi(V_1), \pi(V_2), \cdots, \pi(V_{|\Omega_V|})]$ can be calculated by solving Eq. (2.3) for all network states V_i in Ω_V with a normalizing condition $\sum_i \pi(V_i) = 1$. There is a simple method to solve these equations by forming a transition rate matrix $\boldsymbol{Q}$ of size $|\Omega_V| \times |\Omega_V|$, where its elements q_{it} are obtained by Eq. (2.2), and solving $\boldsymbol{\pi Q} = 0$ subject

to $\sum_i \pi(V_i) = 1$. As we have an extra equation ($\sum_i \pi(V_i) = 1$) than the number of variables $|\Omega_V|$, we can take the transpose of $\boldsymbol{Q}$ matrix as $\boldsymbol{Q}^T$, and replace one of its rows (say, first row) by an array of all ones, which results in another matrix, say $\boldsymbol{A}$. We utilize a LSQR method [61] to obtain the steady network state distribution $\pi(V_i), i = 1, 2, \ldots, |\Omega_V|$ using the following Matlab command: $\boldsymbol{\pi} = \text{LSQR}(\boldsymbol{A}, b, \epsilon, maxit)$, where $b = [1, 0, \cdots, 0]^T$, ϵ is the tolerance for convergence, and *maxit* is the maximum number of iterations allowed to find a solution. Note that some of the state probabilities obtained with the LSQR method under the low network load operation could be very small negative values, so these values need to be discarded by replacing them with 0 or with a very small constant.

The overall exact blocking probability (BP) in an EON with or without SC is given by ensemble averaging over blocking probability (BP_k^o) of all classes $k, k = 1, \ldots, K$ on all OD pair requests $o \in \mathcal{O}$, and per class per OD pair blocking probability BP_k^o is given using the blocking identification function $\mathbb{B}(o, k)$ as

$$BP_k^o = \sum_{i=1, V_i \in \mathbb{B}(o,k)}^{|\Omega_V|} \pi(V_i). \tag{2.4}$$

2.5. Reduced State Model Description

The exact network state model is computationally intractable for medium or large scale links and networks. Thus, it is essential to represent the state of a network that characterizes the occupancy of all links separately or simply by the average occupancy of links. There are mainly two features that one can use to define each link-state in a network: the number of connections per class **n** or total occupied slices x. A reduced Macrostate model can be formulated by defining a link-state with **n**, however, it would still be not very scalable. To deal with the scalability issue, we need to represent the state of a link j by its total occupancy x_j.

2.5.1. Reduced Link-State Representation

Table 2.3 shows how an exact link-state description (formed by a single route) can be equivalently represented by only a few microstates, which represent the corresponding total occupied slices. In this 7-slice link example, there are 15 exact link-states under the RF policy. However, for example, all 5 exact states having total occupancy of 3 slices (S_2 to S_6 in first column) are represented by a microstate $x = 3$, where $X = x = \boldsymbol{n}^1 \cdot \boldsymbol{d}^T = \sum_{k=1}^{K} n_k^1 d_k$, where $\boldsymbol{n}^1 \equiv (n_1^1, \ldots, n_k^1, \ldots, n_K^1)$, and n_k^1 is the number of class k connections of an OD pair request $o = 1$. For example, even in a small scale link with 20 slices and bandwidth demands $\boldsymbol{d} = (3, 4, 5)$ slices, under the RF policy the number of exact link-states is 5885, which could be reduced to 19 with microstates representation. Thus, the reduced state (Microstate) model presents an opportunity to obtain approximate blocking probabilities even for large scale links and networks, since the maximum (for $d_1 = 1$) number of microstates per link is $C + 1$, where C is the number of slices per fiber-link. Note that the term "state" is also used in the context of the models, i.e., a state in the Microstate model has the same meaning as a microstate.

Departing from the exact state representation to a microstate representation causes some inaccuracy in finding the connection setup rates in a reduced link-state model. The reason is that a microstate could be represented by different class-dependent blocking and non-blocking exact link-states formed by one or more routes. For example, a microstate with occupancy $x = 3$ slices is represented by five (S_2 to S_6) different exact link-states out of which S_4 is a blocking state for both classes of demands requiring 3 and 4 consecutive free slices, and S_3 and S_5 are additional blocking states for a 4-slice demand.

Let us first define a set of blocking exact link-states for an incoming class k request in a microstate $X = x$ by Eq. (2.5), which can not admit a demand with d_k slices due to the fact that the size

Table 2.3.: Possible link-state representations with capacity $C = 7$ slices, demands $\boldsymbol{d} = (3, 4)$ slices under the RF policy [58].

Exact link-states (Ω_S)	Macrostates	Microstates
$S = (s_1, s_2, s_3, s_4, s_5, s_6, s_7)$	$\boldsymbol{n}^1 = (n_1^1, n_2^1)$	$X = x = \boldsymbol{n}^1 \cdot \boldsymbol{d}^T$
$S_1 = (0, 0, 0, 0, 0, 0, 0)$	(0,0)	0
$S_2 = (1, \infty, \infty, 0, 0, 0, 0)$ $S_3 = (0, 1, \infty, \infty, 0, 0, 0)$ $S_4 = (0, 0, 1, \infty, \infty, 0, 0)$ $S_5 = (0, 0, 0, 1, \infty, \infty, 0)$ $S_6 = (0, 0, 0, 0, 1, \infty, \infty)$	(1,0)	3
$S_7 = (1, \infty, \infty, \infty, 0, 0, 0)$ $S_8 = (0, 1, \infty, \infty, \infty, 0, 0)$ $S_9 = (0, 0, 1, \infty, \infty, \infty, 0)$ $S_{10} = (0, 0, 0, 1, \infty, \infty, \infty)$	(0,1)	4
$S_{11} = (1, \infty, \infty, 1, \infty, \infty, 0)$ $S_{12} = (1, \infty, \infty, 0, 1, \infty, \infty)$ $S_{13} = (0, 1, \infty, \infty, 1, \infty, \infty)$	(2,0)	6
$S_{14} = (1, \infty, \infty, 1, \infty, \infty, \infty)$ $S_{15} = (1, \infty, \infty, \infty, 1, \infty, \infty)$	(1,1)	7

of the largest consecutive free slices (f_l) on a link is not sufficient. Therefore, a set of non-blocking exact link-states can be given by Eq. (2.6).

$$\mathbb{B}(x, k) = \{S_i | d_k > f_l(S_i),\ S_i \in \Omega_S(x),\ \forall i\} \tag{2.5}$$

$$\mathbb{NB}(x, k) = \Omega_S(x) \setminus \mathbb{B}(x, k) \tag{2.6}$$

The blocking in a link happens either due to insufficient free spectrum, referred to as *resource blocking* or due the fragmentation of free spectrum resources, referred to as *fragmentation blocking*. Fragmented states ($\mathbb{FI}(x, k)$) do have enough free slices, but they are scattered and the largest block of consecutive free slices ($f_l(S_i)$) can

not satisfy demand d_k. Thus, when a spectrum reallocation strategy is not applied to reconfigure connections or a multipath routing (i.e., traffic splitting over multiple paths) is not applied in a network, then fragmented exact states also block arriving connection requests. Thus, a class-dependent set of fragmented exact states corresponding to a microstate $X = x$ is given by Eq. (2.7), and the set of resource blocking states $\mathbb{RB}(x,k) \subseteq \mathbb{B}(x,k)$ is $\mathbb{B}(x,k) \setminus \mathbb{FI}(x,k)$.

$$\mathbb{FI}(x,k) = \{S_i | f_l(S_i) < d_k \leqslant C - x, S_i \in \Omega_S(x), \forall i\} \tag{2.7}$$

The number of elements in $\Omega_S(x), \mathbb{NB}, \mathbb{FI}$, and $\mathbb{RB}$ sets can be obtained using a simple procedure by generating all exact link states under a given spectrum allocation scenario (e.g., RF, FF) using an approach described in Algorithm 1 in Section 2.4 for a small scale single-hop (link) network. However, in medium and large scale links with capacity $C > 20$, an algorithmic approach would not be useful, since the time and space complexity increase exponentially. Thus, using the inclusion-exclusion principle, we provide analytical expressions for computing the number of exact states ($|\Omega_S(x)|$) on a link with r traversing routes in Theorem 1, the number of non-blocking exact link states ($|\mathbb{NB}(x,k)|$) in Theorem 2, and the number of fragmented states ($|\mathbb{FI}(x,k)|$) in Theorem 3, where the number of free slices $E(x) = C - x$ and the number of connections $N(\mathbf{n}) = \sum_{k=1}^{K} \sum_{o=1}^{\mathrm{r}} n_k^o(\mathbf{n})$. The proofs are given in Appendix A. The number of resource blocking exact link states is $|\mathbb{RB}(x,k)| = |\Omega_S(x)| - |\mathbb{NB}(x,k)| - |\mathbb{FI}(x,k)|$.

Theorem 1 *Under the RF policy, the number of exact link-states in a given occupancy state x is*

$$|\Omega_S(x)| = \sum_{\boldsymbol{n} \in \Omega_S(x)} \frac{N(\boldsymbol{n})!}{\prod_{k=1}^{K} \prod_{o=1}^{\mathrm{r}} n_k^o(\boldsymbol{n})!} \times \binom{E(x) + N(\boldsymbol{n})}{N(\boldsymbol{n})}. \tag{2.8}$$

Proof 1 *The proof is given in Appendix A.*

Theorem 2 *Under the RF policy, the number of non-blocking exact link-states for a class k request with demand d_k slices in a given occupancy state x is*

$$|\mathbb{NB}(x,k)| = \sum_{\boldsymbol{n}\in\Omega_S(x)} W(\boldsymbol{n}) \times \frac{N(\boldsymbol{n})!}{\prod_{k=1}^{K}\prod_{o=1}^{\mathrm{r}} n_k^o(\boldsymbol{n})!}, \text{ where} \tag{2.9}$$

$$W(\boldsymbol{n}) = \sum_{i=1}^{N(\boldsymbol{n})+1} (-1)^{i+1} \binom{N(\boldsymbol{n})+1}{i} \binom{E(x)+N(\boldsymbol{n})-id_k}{N(\boldsymbol{n})}.$$

Proof 2 *The proof is given in Appendix A.*

Theorem 3 *For any policy, the number of fragmentation exact link-states ($|\mathbb{FI}(x,k)|$) for a class k request with demand d_k slices in a given occupancy state x is*

$$|\mathbb{FI}(x,k)| = \begin{cases} |\Omega_S(x)| - |\mathbb{NB}(x,k)|, & 0 \leqslant x \leqslant C - d_k \\ 0, & \text{otherwise.} \end{cases} \tag{2.10}$$

Proof 3 *As the set of exact link-states with total occupancy of x slices, i.e., $\Omega_S(x)$ can be divided into disjoint subsets of non-blocking $\mathbb{NB}(x,k)$, fragmentation $\mathbb{FI}(x,k)$ and resource blocking states $\mathbb{RB}(x,k)$ for a class k requests, the following equation holds $|\mathbb{FI}(x,k)| = |\Omega_S(x)| - |\mathbb{NB}(x,k)| - |\mathbb{RB}(x,k)|$. Furthermore, $|\mathbb{RB}(x,k)| = 0$ for $0 \leqslant x \leqslant C - d_k$, and $|\mathbb{RB}(x,k)| = |\Omega_S(x)|$ for $x > C - d_k$. Therefore, the above theorem holds.*

For example, in Table 2.3, the number of exact link-states corresponding to $x = 3$, which is represented by a single route and unique $\mathbf{n}^1 = (1,0)$, is $\frac{1!}{1!} \times \binom{4+1}{1} = 5$. On the other hand, the number of non-blocking exact states for a class 2 demand ($d_k = 4$ slices) in a microstate $x = 3$ i.e., $\mathbf{n}^1 = (1,0)$ and $E(x=3) = 4$ slices, is $\binom{1+1}{1} \times \binom{4+1-4}{1} = 2$, which can be seen in Table 2.3. Notice that the number of exact link-states for a microstate as given by Eq. (2.8) is only valid under the RF spectrum allocation policy, the number of

valid exact link-states under the FF policy is generally much lower. Furthermore, as described in Section 2.4, exact link-states are formed according to a given spectrum allocation policy, classes of demands, and the number of routes that traverses a link under consideration, which further increases the complexity.

We consider all exact model assumptions (in Section 2.4) in the reduced state model, which is based on a multiclass loss network framework [54, 55]. Below we list additional assumptions and definitions of our analytical model.

A1) A network state is defined by a set of all link occupancy states $\{X_j, j \in \mathcal{J}\}$, where X_j represents the number of occupied slices (x_j) on a link j.

A2) Spectrum occupancy in a link j is independent from other links $i \neq j; i, j \in \mathcal{J}$, which is called the statistical *independence link* assumption, i.e., $Pr[X_j = x_j | X_i = x_i] = Pr[X_j = x_j], i \neq j$.

A3) All exact link states are formed by a single route, i.e., r = 1 in Eqs. (2.8) and (3), and $\mathbf{n} \equiv \mathbf{n}^1 = (n_1, \ldots, n_K)$, where the route number is omitted.

A4) The connection acceptance probability computed by the EES (EOV) approach assumes that all exact link (network) states with same link (route) occupancy are equiprobable. The SOC approach, on the other hand, assumes that exact link-state probabilities of non-blocking states are higher than the corresponding blocking states at lower loads. The Uniform approach assumes that spectrum occupancy is uniformly distributed.

It is important to mention that A1–A3 are the reduced state model definitions and assumptions useful for tractability of the model, whose blocking computing methodology is described in Sec. 2.6. As mentioned in Sec. 2.3, we propose various approximation approaches: EES, SOC, EOV and Uniform, for the computation of a

connection acceptance probability term used in the calculation of class- and state-dependent connection setup rate. Our approaches, and existing ones [22], assume partially valid or invalid spectrum occupancy patterns, and consider either partially or ignore the spectrum contiguity and continuity constraints, as shown in Table 2.1. Assumptions specific to each approximation approach listed in A4 are explained along the description of the approach given in the next subsections.

2.5.2. Probability of Acceptance of a Connection in a Link

Let us now use the above assumptions and definitions of non-blocking and blocking states to reduce an exact link-state model into a reduced microstate model. Let a random variable X_j be the number of occupied slices (x_j) on a link j, and the probability that a link j is in state x_j as

$$\pi_j(x_j) \equiv Pr[X_j = x_j] \tag{2.11}$$

Therefore, using the link-state probability obtained for a given load, the average occupied slices on a link j is given by Eq. (2.12), and the average occupied slices on a h-hop route is denoted by a vector $\bar{\mathbf{x}}_r \equiv (\bar{x}_{j_1}, \bar{x}_{j_2}, \cdots, \bar{x}_{j_h})$, where the route $r \equiv \{j_1, j_2, \cdots, j_h\}$.

$$\bar{x}_j = \sum_{0 \leqslant x_j \leqslant C} x_j \pi_j(x_j). \tag{2.12}$$

In the reduced model the transition rate from a microstate $X_j = x_j$ to another microstate due to an arrival of a class k request (i.e., connection setup rate) depends on the connection arrival rate and the probability of its acceptance. Noting that only non-blocking exact states corresponding to the microstate $X_j = x_j$ will accept the incoming request, in a single link system (route $r = \{j\}$) the probability of acceptance of a class k connection request with bandwidth d_k in a given occupancy (microstate) $X = x$ (omitting the subscript

j), i.e., $p_k(x)$ is obtained by

$$p_k(x) \equiv Pr[Z_r \geqslant d_k | X = x] = \sum_{S_i \in \Omega_S(x)} Pr[f_l(S_i) \geqslant d_k | S_i, X = x] \times Pr[S_i | X = x] \quad (2.13)$$

where the event $\{Z_r \geqslant d_k\}$ represents that the route r (here a link j) must have equal or more than d_k consecutive free slices to accept a class k request. In a given microstate $X = x$, only a subset of exact states representing a microstate x that have sufficient consecutive free slices would accept the class k request ($\forall S_i \in \Omega_S(x) : f_l(S_i) \geqslant d_k$). The first multiplication term in Eq. (2.13) is a probability function resulting in a value 1 if an exact state S_i is a non-blocking state, 0 otherwise. The second term is the probability of observing the link in an exact state S_i among the set of exact states representing occupancy of x slices, i.e., $\Omega_S(x)$. However, the calculation of exact state probabilities (for the second term) in a large link is analytically intractable. We need, therefore, some kind of approximation to calculate the class- and state-dependent probability of acceptance and connection setup rates. Assuming that *all exact states corresponding to a given microstate have uniform state probability distribution*, i.e., they are equiprobable. Thus, $Pr[S_i | X = x] = 1/|\Omega_S(x)|, \forall S_i \in \Omega_S(x)$. We refer to this approximation as an equiprobable exact states (EES) approach. As only non-blocking exact states ($\mathbb{NB}(x,k)$) would allow a class k connection to be accepted in a microstate x, therefore, the first multiplication term in Eq. (2.13) would add up to the total number of exact non-blocking states ($|\mathbb{NB}(x,k)|$) representing an occupancy x. Thus, the probability of acceptance in Eq. (2.13) can be approximated in the EES approach as

$$p_k^{App.EES}(x) = \frac{|\mathbb{NB}(x,k)|}{|\Omega_S(x)|}. \quad (2.14)$$

The state-dependent per-class connection setup rate in a link is

given as the class k arrival rate ($\lambda_k \equiv \lambda_k^o$) multiplied by the probability of acceptance of an incoming connection request with demand d_k slices in a microstate $X = x$, i.e., $\alpha_k(x) = \lambda_k \times p_k^{App.EES}(x)$.

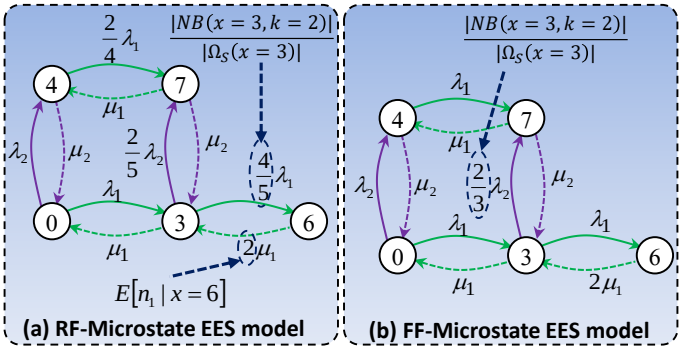

Figure 2.5.: Microstate transition diagrams of a 7-slice fiber link with two classes of demands $\boldsymbol{d} = (3, 4)$ slices under RF in (a) and FF in (b) are shown [58].

To illustrate the transitions and connection setup rates in a reduced link state model using the EES approximation, let us consider an example in Fig. 2.5, where the microstate transition diagram of a 7-slice link occupancy is shown with two classes of demands $\boldsymbol{d} = (3, 4)$ slices under the RF policy in Fig. 2.5(a), and under the FF policy in Fig. 2.5(b). As can be seen in Fig. 2.5(a), the overall connection setup rate in the empty microstate $x = 0$ is λ_1 and λ_2 for class-1 (3-slice) and class-2 (4-slice) connection request, respectively, since the corresponding exact empty state S_1 is a non-blocking state for both connection classes, under both RF and FF policies. However, in a microstate $x = 3$, which represents 5 different exact states of $\boldsymbol{n} = (1, 0)$ in the RF policy, four (two) states are non-blocking for class $k = 1 (k = 2)$, see Table 2.3. Using the EES, the connection setup rate for class-1 (class-2) in the microstate

$x = 3$ is $\frac{4}{5}\lambda_1$ ($\frac{2}{5}\lambda_2$). In contrast, in the FF policy, which allocates only first available slices, generally generates lesser number of exact states as compare to the RF policy. In this example, in the FF policy a microstate $x = 3$ is represented by only three exact states $(1, \infty, \infty, 0, 0, 0, 0), (0, 0, 0, 1, \infty, \infty, 0)$ and $(0, 0, 0, 0, 1, \infty, \infty)$ due to (de)allocation processes, out of which there is only a single exact state $(0, 0, 0, 1, \infty, \infty, 0)$ that blocks a class-2 demand. Thus, using Eq. (2.14) the class-2 connection setup rate in the microstate $x = 3$ is $\alpha_2(x = 3) = \frac{2}{3}\lambda_2$ as shown in Fig 2.5(b). Notice that the transition rate from a state occupancy $x = 6$ to $x = 3$ is $2\mu_1$, since the transition occurs due to the departure of a class-1 connection (3 slices bandwidth), and the expected number of class-1 connections in $x = 6$ is 2 in both policies, because $x = 6$ is represented by only one $\boldsymbol{n} = (2, 0)$ as shown in Table 2.3.

To test the EES assumption corresponding to each microstate (total occupied slices) under both spectrum allocation policies, we plot the variance of exact state probabilities using the RF and the FF exact link models against total occupied slices in Fig. 2.6. We can see that the variance is non-zero for all microstates, which should not have been the case for the reduced state model to be accurate. Nevertheless, the variance under the RF policy is not as high as the variance obtained in the FF policy. Thus, we make the following observations.

Observation 1: The probability of observing a microstate (x) in its exact states is not eqiprobable. The variance depends on the occupancy and load, or equivalently, average occupancy $\bar{x}$.

Observation 2: In a lower microstate ($x \ll \bar{x}$) where the variance is higher, the probability of observing its non-blocking exact states is more likely than observing its blocking exact states.

Noting the observations 1 and 2, and the fact that a class k request with demand d_k slices could be accepted in microstates $0 \leqslant x \leqslant C - d_k$, which are represented by non-blocking and frag-

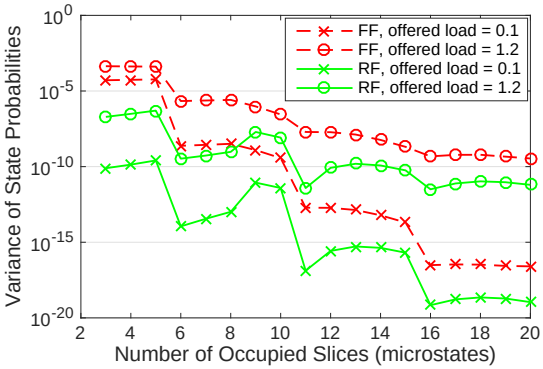

Figure 2.6.: Variance of steady state probabilities of a set of exact states representing a link occupancy (microstate) for $C = 20, \boldsymbol{d} = (3, 4, 5)$ [58].

mentation exact states, we propose a load- and state-dependent SOC approximation. Let us assume that for any class k request in a microstate x, the corresponding non-blocking exact states are equiprobable, so are the fragmentation exact states. However, for lower microstates $x < \bar{x}$, the probability of observing a non-blocking exact state is higher than observing a fragmentation exact state in a given microstate (x). Therefore, let us assume that the probability of observing a non-blocking exact state in a microstate x is in the form $a = \frac{1}{|\Omega_S(x)|}[1 + u\exp(-n)]$, and for a fragmentation exact state, it is $b = \frac{1}{|\Omega_S(x)|}[1 - v\exp(-n)]; 0 \leqslant a, b \leqslant 1$. Additionally, in a given microstate $x, 0 \leqslant x \leqslant C - d_k$, $a \times |\mathbb{NB}(x,k)| + b \times |\mathbb{FI}(x,k)| = 1$, and $|\Omega_S(x)| = |\mathbb{NB}(x,k)| + |\mathbb{FI}(x,k)|$. Also noting that only non-blocking states would accept an incoming class k request, a load-dependent $(\bar{x})$ approximate (SOC) probability of acceptance of a request in a

link may be given by Eq. (2.15).

$$p_k^{App.SOC}(x;\bar{x}) = \frac{|\mathbb{NB}(x,k)|}{|\Omega_S(x)|} + \frac{|\mathbb{FI}(x,k)|}{|\Omega_S(x)|}\exp(-\frac{\bar{x}}{C} \times |\ln(\frac{x}{\bar{x}})|) \quad (2.15)$$

Notice that $0 \leqslant p_k^{App.SOC}(x;\bar{x}) \leqslant 1$, since $|\Omega_S(x)| = |\mathbb{NB}(x,k)| + |\mathbb{FI}(x,k)|$ for $0 \leqslant x \leqslant C - d_k$, and $p_k^{App.SOC}(x;\bar{x}) = 0$ for resource blocking states, i.e., $C - d_k < x \leqslant C$, since $|\mathbb{NB}(x,k)| = |\mathbb{FI}(x,k)| = 0$, and $|\Omega_S(x)| = |\mathbb{RB}(x,k)|$. As a special case, the probability of acceptance of a class k request with demand $d_k \leqslant C$ in an empty state is defined as $p_k^{App.SOC}(0;\bar{x}) = 1$, since $|\Omega_S(0)| = |\mathbb{NB}(0,k)| = 1$, and $|\mathbb{FI}(0,k)| = 0$. The difference between the EES and SOC is the second factor in Eq. (2.15), which increases the probability of acceptance for lower occupancy states x, and decreases it for higher states depending on the average occupied slices ($\bar{x}$) for a given load. At the same time, the computation of acceptance probability in Eq. (2.15) requires the knowledge of $\bar{x}$, and vice versa, which makes it a coupled equation. Thus, $\bar{x}$ needs to be computed using an iterative procedure described in Section 2.6.

2.5.3. Probability of Acceptance of a Request on a Multi-hop Route

Let Z_r be the random variable representing the size of the largest continuous and contiguous free slices on a route $r = \{j_1, j_2, \cdots, j_h\}$ without SC, where j_i represents a link on the route r. In contrast, under the SC operations, Z_r would represent the minimum of the largest size of contiguous free slices on each constituent link of route r, i.e, $Z_r = \min(Z_{j_1}, \ldots, Z_{j_h})$, where Z_{j_i} is a random variable representing the size of the largest contiguous free slices on a link j_i, and the Z_{j_i}'s are statistically independent due to the independence link assumption A1 in Section 2.5.1. The probability of acceptance of a connection request with demand d_k slices on a route r without SC is given in Eq. (2.16), which extends the single-hop approach

in Eq. (2.13) to an h-hop route in a given route occupancy vector $\mathbf{x}_r = (x_{j_1}, \ldots, x_{j_h})$ with a parameter average route occupancy vector $\bar{\mathbf{x}}_r = (\bar{x}_{j_1}, \bar{x}_{j_2}, \cdots, \bar{x}_{j_h})$.

$$p_k(\mathbf{x}_r; \bar{\mathbf{x}}_r) \equiv Pr(Z_r \geqslant d_k \mid X_{j_1} = x_{j_1}, ..., X_{j_h} = x_{j_h}; \bar{\mathbf{x}}_r)$$
$$= \sum_{V_t \in \Omega_V(\mathbf{x}_r)} Pr[f_l(\cup_{i=1}^{h} S_t^{j_i}) \geqslant d_k | V_t, \mathbf{x}_r; \bar{\mathbf{x}}_r] \times Pr[V_t | \mathbf{x}_r; \bar{\mathbf{x}}_r] \quad (2.16)$$

Here, the analogy is similar to that of a single-hop, i.e., in place of an exact link-state S_i, now a set of h exact link-states on the route r, belonging to a network state $V_t \in \Omega_V(\mathbf{x}_r)$, i.e., $(S_t^{j_1}, \ldots, S_t^{j_h})$, $S_t^{j_i} \in \Omega_S(x_{j_i})$ determines whether this set of exact link-states has equal or more than d_k free contiguous and continuous slices or not, which is given by the first multiplication term in Eq. (2.16) with probability 1 or 0. Notice that a slice-wise union operation over a set of h exact link-states finds the number of aligned free slices, i.e., fulfilling the continuity constraint, and the function $f_l(\cdot)$ finds the largest free contiguous slices among the aligned free slices. Similar to the single-hop analysis, the probability of observing a network state $V_t \in \Omega_V$ in a given route occupancy vector $\mathbf{x}_r$ requires exact network state probabilities. Thus, if we again assume that the network states with route occupancy $\mathbf{x}_r$ are equiprobable and also load-independent, then the second probability term is approximated as $1/|\Omega_V(\mathbf{x}_r)|$. However, unlike the single-hop system, an analytical expression for computing the number of such non-blocking network states that could accept a request with demand d_k (i.e., first summation term) is rather difficult, if not impossible. Thus, a procedure (Algorithm 2) to compute the probability of acceptance using a load-independent *equiprobable occupancy vector* (EOV) approach is given without SC below. We refer this approximation as *App.EOV*. Notice that Algorithm 2 requires all exact link-states, not the network states, to be generated for a given spectrum allocation policy. Thus, the number of network states with route occupancy $\mathbf{x}_r$ is approxi-

mated as $|\Omega_V(\mathbf{x}_r)| \approx \prod_{i=1}^{h} |\Omega_S(x_{j_i})|$.

Algorithm 2 Compute $p_k^{App.EOV}(x_{j_1}, x_{j_2}, x_{j_3}, \cdots, x_{j_h})$.

1: **Given**: Demand d_k; Route occupancy $\mathbf{x}_r$; Link-states $\Omega_S(x_{j_i}), i = 1, 2, \ldots, h$.
2: Initialize $nb \leftarrow 0$.
3: **if** $d_k \leq C - x_{j_i},\ \forall j_i \in r$ **then**
4: **for** each link-state S^{j_1} in $\Omega_S(x_{j_1})$ **do**
5: **for** each link-state S^{j_2} in $\Omega_S(x_{j_2})$ **do**
6: $\vdots$
7: **for** each link-state S^{j_h} in $\Omega_S(x_{j_h})$ **do**
8: $nb++$, if $f_l(\cup_{i=1}^{h} S^{j_i}) \geq d_k$
9: **end for**
10: **end for**
11: **end for**
12: **end if**
13: $p_k^{App.EOV}(x_{j_1}, x_{j_2}, x_{j_3}, \cdots, x_{j_h}) = \frac{nb}{\prod_{i=1}^{h} |\Omega_S(x_{j_i})|}$

In Algorithm 2, a slice-wise union operation over a set of all possible combination of exact link-states satisfying the occupancy on each link of an h-hop route together with a function $f_l(\cdot)$ is used in step 8 to find the number of non-blocking route occupancy states. For example, the union of occupancies on link 1 $S^{j_1} = (0, 1, \infty, \infty, 0, 0, 0)$ and on link 2 $S^{j_2} = (0, 0, 1, \infty, \infty, 0, 0)$ is $\cup_{i=1}^{2} S^{j_i} = (0, 1, \infty, \infty, \infty, 0, 0)$. Thus, a 2-link path in the route occupancy state (S^{j_1}, S^{j_2}) can accept a 2-hop request with demand $0 < d_k \leq 2$, since the largest number of consecutive free slices $f_l(0, 1, \infty, \infty, \infty, 0, 0) = 2$. The normalized number of all favorable route occupancy states gives the probability of acceptance in step 13. Notably, we can use Algorithm 2 to find $p_k^{App.EOV}(x)$ for any given policy (e.g., RF, FF) provided the exact link-states are precomputed.

However, the EOV approximation approach is useful only for small scale networks with link capacity $C \leqslant 10$, since generating all exact link-states is computationally intractable for a network with higher link capacity.

To address the scalability issue of the EOV approach, we present load-independent analytical expressions for computing the probability of acceptance term without SC in Eq. (B-3) and with SC in Eq. B-4 (see Appendix B) that assume a uniform distribution of total occupancy over spectrum slices, i.e., spectrum patterns (link-states) are created only by a single-slice demand. We refer this approach as a *Uniform* approximation (denoted by App.Uni). The *Uniform* approximation considers required RSA constraint(s) in with and without SC operations, it ignores the valid spectrum patterns and fragmentation created due to bandwidth demands.

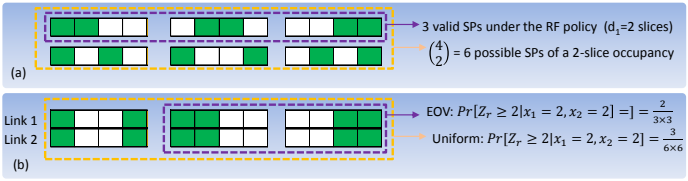

Figure 2.7.: (a) Various 2-slice ($x = 2$) occupancy patterns on a link with capcity 4 slices. Free and occupied slices are shown in white and green, respectively. (b) Favorable combination of occupancy on a 2-hop route to accept a demand of 2 consecutive slices: in the EOV approach under RF policy, and in the Uniform approach.

Let us consider an example in Fig. 2.7 to illustrate how some of the factors like valid/invalid state representation, i.e., spectrum patterns (SPs) formed by a set of bandwidth demands with the given RSA policy and constraints is used to compute the probability of acceptance under the EOV and Uniform approximation approaches.

Fig. 2.7(a) shows how a two-slice occupancy is represented in a link with capacity $C = 4$ slices in all 6 possible ways, i.e., with uniform distribution. The same link occupancy is represented by 3 valid SPs under the RF policy with demand $d_1 = 2$ consecutive slices. On the other hand, only a set of combination of spectrum patterns on a 2-hop route in a two-slice occupancy per link j ($x_j = 2, j = 1, 2$) accept a new two-slice bandwidth request over two contiguous and continuous free slices (Fig. 2.7(b)) with different acceptance probabilities (i.e., $Pr[Z_r \geqslant 2|x_1 = 2, x_2 = 2]$) by different approaches. Here, only 2 (3) out of 9 (36) combination of occupancy patterns on a 2-link path with two occupied slices per link could accept a demand of 2 consecutive slices under the RF policy in the EOV (Uniform) approach. If we consider the EOV approach as a benchmark, the Uniform approach underestimates the computation of probability of acceptance term due to the assumption of uniform distribution of occupancy.

Achieving scalability and accuracy (using valid spectrum patterns and RSA constraints) at the same time is hard. Nevertheless, utilizing the independence link assumption A1 in Section 2.5.1, and noting that a product-form approximation is also a valid probability distribution [21, 62], the approximate probability of acceptance of a request on a route with h hops in an EON without SC may be given by a product of all individual probability of acceptance in constituent links on each hop of an h-hop route r as below.

$$p_k^{App.}(\mathbf{x}_r; \bar{\mathbf{x}}_r) = \left[\prod_{i=1}^{h} Pr(Z_{j_i} \geqslant d_k | X_{j_i} = x_{j_i}; \bar{x}_{j_i}) \right]^h \tag{2.17}$$

Notice that the individual link acceptance probability term $Pr(Z_{j_i} \geqslant d_k|X_{j_i} = x_{j_i}; \bar{x}_{j_i})$ finds the probability that the route r has equal or more than d_k free consecutive slices on each link $j_i \in r$ with occupancy x_{j_i}. Using the independent link assumption, the probability that a route r with occupancy $\mathbf{x}_r$ has at least d_k consecutive free

slices on each constituent link is given by the product of constituent link acceptance probability terms, i.e., $\prod_{i=1}^{h} Pr(Z_{j_i} \geqslant d_k | X_{j_i} = x_{j_i}; \bar{x}_{j_i})$. However, it does not necessarily ensure that the contiguous free slices are aligned over the route r, i.e., the continuity constraint. Therefore, the effect of the spectrum continuity constraint in Eq. (2.17) may be partially taken into account by considering only a fraction (using power h) of all route occupancy patterns that have equal or more than d_k consecutive slices on each link of a route r.

The probability of acceptance of a request with demand d_k on an h-hop route with SC, given the route occupancy vector $\mathbf{x}_r$ and average occupancy vector $\bar{\mathbf{x}}_r = (\bar{x}_{j_1}, \bar{x}_{j_2}, \cdots, \bar{x}_{j_h})$ is accurately given as

$$p_{k,sc}(\mathbf{x}_r; \bar{\mathbf{x}}_r) \equiv \prod_{i=1}^{h} Pr(Z_{j_i} \geqslant d_k | X_{j_i} = x_{j_i}; \bar{x}_{j_i}) \tag{2.18}$$

which uses the definition of the random variable Z_r under the SC operations, i.e., $Z_r = \min(Z_{j_i}, \cdots, Z_{j_h})$, and Z_{j_i}'s are independent due to the independence link assumption A1 in Section 2.5.1, so $Pr(\min(Z_{j_1}, \cdots, Z_{j_h}) \geqslant d_k) = \prod_{i=1}^{h} Pr(Z_{j_i} \geqslant d_k)$.

For the EES approach, using Eqs. (2.14) and (2.17) we can obtain the approximate probability of acceptance of a request on a route r with h hops (without SC) by Eq. (2.19). Similarly, for the SC operation modes, using Eqs. (2.14) and (2.18), the probability of acceptance is given on a route r with h hops can be obtained by Eq. (2.20).

$$p_k^{App.EES}(\mathbf{x}_r) = \prod_{i=1}^{h} \left[\frac{|\mathbb{NB}(x_{j_i}, k)|}{|\Omega_S(x_{j_i})|} \right]^h \tag{2.19}$$

$$p_{k,sc}^{App.EES}(\mathbf{x}_r) = \prod_{i=1}^{h} \frac{|\mathbb{NB}(x_{j_i}, k)|}{|\Omega_S(x_{j_i})|} \tag{2.20}$$

It should be noted that the probability of acceptance in the EOV approach under the SC operations can also be computed by Algorithm 2 by replacing the condition in step 8, as it needs to consider only the contiguity constraint on each link, i.e., $f_l(S^{j_i}) \geq d_k, \forall i = 1, 2, \ldots, h$. Also, under the SC operations the EOV and EES approaches will result the same probability of acceptance. The reason is that when the continuity constraint is relaxed in SC operations, the number of non-blocking route occupancy states in the EOV approach can be computed by $\prod_{i=1}^{h} |\mathbb{NB}(x_{j_i}, k)|$, thus, $p_{k,sc}^{App.EOV}(\mathbf{x}_r)$ is also given by Eq. (2.20). Therefore, in the RF-SC scenario, we can expect that approximate BPs obtained by the EES approach will match the exact or simulation results irrespective of the scale of EONs.

The load-dependent approximate probability of acceptance (App.SOC) of a class k request on an h-hop route r without SC is obtained by Eq. (2.21) (using Eqs. (2.15) and (2.17)). Similarly, under the SC operation, it is given by Eq. (2.22) using Eqs. (2.15) and (2.18).

$$p_k^{App.SOC}(\mathbf{x}_r; \bar{\mathbf{x}}_r) = \prod_{i=1}^{h} \left[p_k^{App.SOC}(x_{j_i}; \bar{x}_{j_i}) \right]^h \tag{2.21}$$

$$p_{k,sc}^{App.SOC}(\mathbf{x}_r; \bar{\mathbf{x}}_r) = \prod_{i=1}^{h} p_k^{App.SOC}(x_{j_i}; \bar{x}_{j_i}) \tag{2.22}$$

To show the accuracy of various approximation approaches, we plot the average probability of acceptance $p(x) = \frac{1}{K} \sum_k p_k(x)$ considering all classes in a single-hop link system (i.e., $h = 1$) under the RF policy in Fig. 2.8 and the FF policy in Fig. 2.9. The exact probability of acceptance in Eq. (2.13)) is used as a benchmark, and it requires the knowledge of exact state probabilities. App.EES and App.SOC approximations use Eqs. (2.14) and (2.15), respectively. The Uniform approximation (shown as App.Uni) computes the $p_k(x)$

using Eq. (B-3). Additionally, we plot the average probability of acceptance for two other approximations based on the Kaufman's and the Binomial's approaches presented in [22].

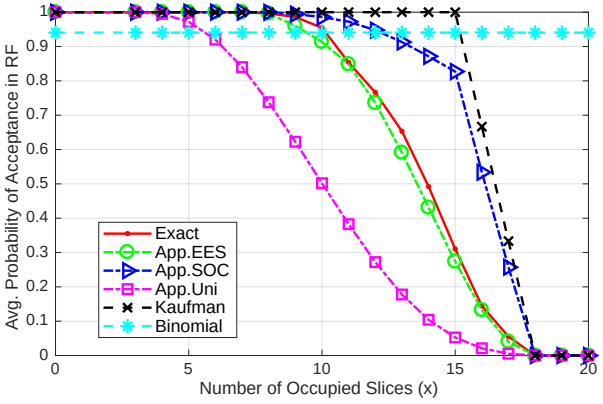

Figure 2.8.: The average probability of acceptance of connection requests under various approaches against the number of occupied slices (microstates) for $C = 20; \boldsymbol{d} = (3, 4, 5)$, and offered load = 1.2.

Notice that the Uniform, Kaufman, and Binomial approaches do not provide different $p(x)$ for the RF and the FF policies, as they do not consider the spectrum patterns formed under a particular RSA. From both the figures, we can say that the EES and SOC approximations are a good match for the exact probability of acceptance under the RF and FF policies, respectively. Thus, we expect that the EES and the SOC approximations will be suitable to obtain approximate blocking probabilities under the RF and the FF scenarios, respectively. Additionally, we observe that the Kaufman's approach over-estimates the probability of acceptance, as it ignores both the RSA constraints (see Eq. 2.23). On the other hand, the Uniform ap-

proximation underestimates the $p_k(x)$, since it assumes the uniform distribution of occupied slices, even though the RSA constrains are considered in computing $p_k(x)$.

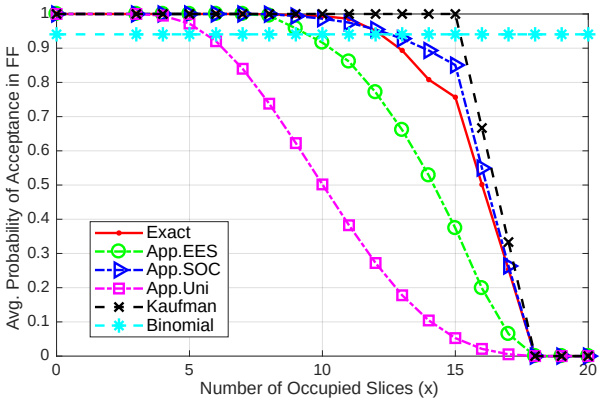

Figure 2.9.: The average probability of acceptance of connection requests under various approaches against the number of occupied slices (microstates) for $C = 20; \boldsymbol{d} = (3, 4, 5)$, and offered load = 1.2.

The Binomial's approach presented in [22], on the other hand, shows a constant probability of acceptance, as it considers the average load (or $\bar{x}$, denoted by ρ in [22]) in order to find the probability of d_k continuous and contiguous free slices on a single-hope ($h = 1$) route r with the link capacity C slices using Eq. (2.24). It is important to mention that there are typos in equations 17, 18, and 21 on page 1627 related to the Binomial approach, i.e., App.2 in [22]. Notice also that the EOV approach is not shown, as this approach will produce the same results for the probability of acceptance as

the EES approach in a single-hop link system.

$$p_k^{Kaufman}(\mathbf{x}_r) = \begin{cases} 1 & \text{if } d_k \leqslant C - x_j, \forall j \in r, \\ 0 & \text{otherwise} \end{cases} \tag{2.23}$$

$$p_k^{Binomial}(C, d_k) = \sum_{x=1}^{d_k} \left[p_k^{Binomial}(C-x, d_k)(1-\rho^h)^{x-1}\rho^h \right] + (1-\rho^h)^{d_k}, \tag{2.24}$$

where $\rho = \frac{1}{C}\sum_{x=0}^{C} x\pi(x)$, and the state probability distribution $\pi(\cdot)$ is computed by the Kaufman's one-dimensional recursive relation, as given below.

$$x \times \pi(x) = \sum_{k=1}^{K} d_k \left(\frac{\lambda_k}{\mu_k}\right) \pi(x - d_k), \pi(x) = 0 \text{ for } x < 0 \text{ and } \sum_{x=0}^{C} \pi(x) = 1 \tag{2.25}$$

2.6. Computing Approximate Network-Wide Blocking

In this section, we present a methodology, as adopted for EONs from the known reduced load approximation model of circuit-switched optical networks [54, 55], to compute approximate blocking probabilities in EONs with or without spectrum conversion.

2.6.1. Calculating Connection Setup and Departure Rates

Generally, the class k connection setup rate in a given link-state is a function of the given link-state occupancy, demand class, and RSA policy [55]. However, links carry different traffic in EONs, thus taking the average occupied slices ($\bar{x}_j$) into consideration, and assuming that the time until the next connection is setup on a link j with x_j occupied slices is exponentially distributed with parameter $\alpha_k^j(x_j)$ [63], the connection setup rate is given by

$$\alpha_k^j(x_j) = \sum_{m=1}^{\kappa_o} \sum_{o:j \in r(o,m)} \lambda_k^{o,m} Pr(Z_{r(o,m)} \geqslant d_k | X_j = x_j; \bar{x}_j). \tag{2.26}$$

Eq. (2.26) calculates the effective arrival rates of all OD pairs $o \in \mathcal{O}$ whose routes $r(o,m), m \geqslant 1$ pass through the link j. Furthermore, the arrival rate on an m^{th} shortest path of an OD pair o is defined as $\lambda_k^{o,m} = \lambda_k^o, m = 1$; and $\lambda_k^{o,m} = \lambda_k^o \times \prod_{u=r(o,1)}^{r(o,m-1)} Pr(Z_u < d_k), m \geqslant 2$. Note that, a request (o,k) is attempted to be routed on its m^{th} shortest path when other link-disjoint shortest paths with index $1,\ldots,m-1$ do not have sufficient free slices to accept a class k request with demand d_k slices. Thus, the overflow arrival rate on a route $r(o,m), m \geqslant 2$ is given by the class k net arrival rate per OD pair (λ_k^o) times the product of individual blocking probability on each preceding route $Pr(Z_u < d_k)$, since routes $\in R(o)$ are link-disjoint. It should also be noted that the effective (reduced) load contribution of an OD pair route $r(o,m)$ on the link j is considered by a probability function $Pr(Z_{r(o,m)} \geqslant d_k | X_j = x_j; \bar{x}_j)$, which depends on the availability of at least required (d_k) free slices (that fulfils the scenario-based RSA constraints) on its route $r(o,m), j \in r(o,m)$ given that link j has x_j occupied slices, and the average occupied slices on link j is $\bar{x}_j$. For notational simplification, we use r in place of $r(o,m)$ whenever possible. Let us consider a 2-hop route $r = \{j_1 = j, j_2\}$. Then, the probability terms are given as

$$Pr[Z_r < d_k] = 1 - Pr[Z_r \geqslant d_k] = 1 - \sum_{x_j=0}^{C-d_k} Pr[Z_r \geqslant d_k | X_j = x_j; \bar{x}_j] \tag{2.27}$$

$$Pr[Z_r \geqslant d_k | X_j = x_j; \bar{x}_j] =$$

$$\sum_{x_{j_2}=0}^{C} Pr[Z_r \geqslant d_k | X_j = x_j, X_{j_2} = x_{j_2}; \bar{x}_j, \bar{x}_{j_2}] \times [X_{j_2} = x_{j_2} | X_j = x_j]$$

$$= \sum_{x_{j_2}=0}^{C-d_k} \pi_{j_2}(x_{j_2}) \times Pr[Z_r \geqslant d_k | X_j = x_j, X_{j_2} = x_{j_2}; \bar{\mathbf{x}}_r].$$

Due to the independence link assumption A1, the random variables X_j and X_{j_2} are independent, so $Pr[X_{j_2} = x_{j_2}|X_j = x_j] = Pr[X_{j_2} = x_{j_2}] = \pi_{j_2}(x_{j_2})$. In general, the above probability term can be calculated for an OD pair traversing route $r = \{j_1 = j, j_2, ..., j_h\}$ with h hops using Eq. (2.28).

$$Pr(Z_r \geqslant d_k|X_j = x_j; \bar{x}_j) = \sum_{x_{j_2}=0}^{C-d_k} \cdots \sum_{x_{j_h}=0}^{C-d_k} \pi_{j_2}(x_{j_2}) \cdots \pi_{j_h}(x_{j_h}) \times p_k(\mathbf{x}_r; \bar{\mathbf{x}}_r) \tag{2.28}$$

In Eq. (2.28), the term after multiplication is defined as the probability of acceptance of a connection with demand d_k, on a route $r(o, m)$ i.e., $p_k(\mathbf{x}_r; \bar{\mathbf{x}}_r) \equiv Pr(Z_r \geqslant d_k|X_j = x_j, X_{j_2} = x_{j_2}, \cdots, X_{j_h} = x_{j_h}; \bar{\mathbf{x}}_r)$. Notice that we used r in place of $r(o, m)$ for simplicity. Therefore, the probability of acceptance on each unique route, whether it is a primary or an alternate route between an OD pair o can be approximately given under various scenarios with and without SC by the EES, SOC, EOV and Uniform approximation approaches, as shown in Section 2.5.3. Additionally, Eq. (2.28) can be simplified to a product of sum expression using Eqs. (2.19)–(2.22) under the EES and SOC approximation, which make them scalable and useful for larger hops networks. For example, under without SC scenarios in the EES approach $Pr(Z_r \geqslant d_k|X_j = x_j; \bar{x}_j) = [p_k^{App.}(x_j; \bar{x}_j)]^h \times \prod_{i=2}^{h} \sum_{x_{j_i}=0}^{C-d_k} \pi_{j_i}(x_{j_i})[p_k^{App.}(x_{j_i}; \bar{x}_{j_i})]^h$.

The expected departure rate of a class k connection in a state x_j is approximately obtained by Eq. (2.29)

$$\begin{aligned} \gamma_k^j(x_j) &= \mu_k \times E[n_k|X_j = x_j] \\ &= \mu_k \times \frac{1}{|\boldsymbol{n}(x_j)|} \sum_{\boldsymbol{n}:\boldsymbol{d}\cdot\boldsymbol{n}^T = x_j} n_k(\boldsymbol{n}) \end{aligned} \tag{2.29}$$

where $E[n_k|X_j = x_j]$ is the expected number of class k connections in the state x_j, which is given by assuming all $\boldsymbol{n}$ that results into the same x_j (i.e., $\boldsymbol{n}(x_j)$) have uniform distribution $\frac{1}{|\boldsymbol{n}(x)|}$, and $n_k(\boldsymbol{n})$ is

the number of class k connections in $\boldsymbol{n}$ (remember the assumption A2 in Section 2.5.1 in the reduced state model, $\boldsymbol{n} \equiv \boldsymbol{n}^1 = (n_1, \ldots, n_k)$).

2.6.2. Computing Approximate Blocking in EONs

Before we calculate blocking probability in EONs, we need to find out the steady state link occupancy distribution $\pi_j(x_j), 0 \leqslant x_j \leqslant C$ for all links $j \in \mathcal{J}$, which can be obtained by solving a set of global balance equations (GBEs) with a normalizing condition $\sum_{x_j=0}^{C} \pi_j(x_j) = 1$ for each link $j \in \mathcal{J}$. The GBE of a valid microstate $X_j = x_j$ is given as

$$\sum_{k=1}^{K} \left(\alpha_k^j(x_j) + \gamma_k^j(x_j)\right) \pi_j(x_j) = \sum_{k=1}^{K} \alpha_k^j(x_j - d_k)\pi_j(x_j - d_k) + \sum_{k=1}^{K} \gamma_k^j(x_j + d_k)\pi_j(x_j + d_k) \quad (2.30)$$

where $\pi_j(i) = 0, \alpha_k^j(i) = 0, \gamma_k^j(i) = 0$ for $i < 0$ and $i > C$. The LHS represents an output flow rate from a microstate $X_j = x_j$ taking into account the connection setup rate $\alpha_k^j(x_j)$ and departure of connection(s) with expected rate $\gamma_k^j(x_j)$, while the RHS represents an input flow rate into the microstate x_j from other state(s) $x_j - d_k$ $(x_j + d_k)$ due to an arrival (departure) of a class k connection demand of d_k slices. Thus, for example, using Eq. (2.30) the GBE of a microstate $x = 3$ in a link in Fig. 2.5(a) can be written as $(\frac{4}{5}\lambda_1 + \frac{2}{5}\lambda_2 + \mu_1)\pi(x = 3) = \lambda_1\pi(x = 0) + 2\mu_1\pi(x = 6) + \mu_2\pi(x = 7)$. Here, the LHS takes into the account of a 3-slice (4-slice) demand arrival in $x = 3$ with effective connection setup rate $4\lambda_1/5$ $(2\lambda_2/5)$. The RHS terms, on the other hand, are due to a class 1 arrival in $x = 0$, and a class 1 and 2 departures in states $x = 6$ and $x = 7$, respectively. Using the LSQR method [61], we can obtain the steady link-state occupancy distribution $\pi_j(x_j)$ for all valid microstates $0 \leqslant x_j \leqslant C$ and links $j \in \mathcal{J}$. Notice that the connection

setup rate ($\alpha_k^j(x_j)$) depends on state probabilities ($\pi_j(x_j)$), and vice versa. Thus, an iterative procedure is required to obtain $\pi_j(x_j)$ and $\alpha_k^j(x_j)$.

The blocking probability of class k bandwidth requests on an OD pair o with bandwidth d_k in an EON can be given as a product of blocking probability of (o,k) requests on each independent link-disjoint route $r \in R(o)$ using Eqs. (2.27) and (2.28) as follows.

$$BP_k^o = \prod_{r \in R(o)} Pr[Z_r < d_k] = \prod_{r \in R(o)} [1 - Pr[Z_r \geqslant d_k]]$$
$$= \prod_{r \in R(o)} [1 - \sum_{x_j=0}^{C-d_k} \sum_{x_{j_2}=0}^{C-d_k} ... \sum_{x_{j_h}=0}^{C-d_k} (\pi_j(x_j) \cdots \pi_{j_h}(x_{j_h}) \times p_k(\mathbf{x}_r; \bar{\mathbf{x}}_r))] \tag{2.31}$$

It should be noted that we use Eq. (2.31) to obtain approximate blocking probabilities in an EON with and without SC, by calculating $p_k(\mathbf{x}_r; \bar{\mathbf{x}}_r)$ separately for with and without SC scenarios under different approximation approaches.

2.6.3. Fixed-Point Iteration for Computing Approximate Blocking Probabilities in EONs

The calculation of approximate blocking probability per class per OD pair (BP_k^o) requires the information of steady state link occupancy probabilities ($\pi_j(x_j)$) of each traversed link of routes in $R(o)$; and these probabilities $\pi_j(x_j)$ can be obtained by solving the nonlinear coupled equations in Eq. (2.30). However, these equations, which are a function of $\boldsymbol{\alpha}$ and $\boldsymbol{\pi}$, could be made linear by a fixed-point iterative procedure (repeated substitution) as follows [55].

1) For all classes $k \in \{1, 2, \cdots, K\}$ and OD pairs $o \in \mathcal{O}$, initialize blocking probabilities $\hat{BP}_k^o = 0$, and for each link $j \in \mathcal{J}$ set $\alpha_k^j(x_j) = \sum_{o:j \in r(o)} \lambda_k^o \times \frac{|\mathbb{NB}(x_j,k)|}{|\Omega(x_j)|}$.

2) Determine the link-state occupancy distribution for valid $x_j, 0 \leqslant x_j \leqslant C$ as $\boldsymbol{\pi_j} = [\pi_j(x_j = 0), \ldots, \pi_j(x_j = C)]$ for each link $j \in \mathcal{J}$ by solving equations Eq. (2.30) and $\sum_{x_j=0}^{C} \pi_j(x_j) = 1$.

3) Calculate $\bar{x}_j \ \forall j \in \mathcal{J}$ by Eq. (2.12), and per-class connection setup rate $\alpha_k^j(\cdot) \ \forall j, \forall k$ using Eq. (2.26).

4) Calculate BP_k^o for all routes and for all classes k by Eq. (2.31).

5) If $\max_{o,k} |\hat{BP}_k^o - BP_k^o| < \epsilon$ then terminate. Else, let $\hat{BP}_k^o = BP_k^o$ and go to step (2).

2.7. Numerical and Simulation Results

In this section, we investigate the effectiveness of the exact and approximate models in obtaining accurate blocking probabilities (BPs) results in small networks, 2-link and 3-node networks as shown in Figs. 2.2(a) and 2.2(b), respectively. Additionally, we study the approximate models (EES and SOC) in larger size networks, including 10-link, 14-node (42 links) NSFNET and 15-node Atlanta topologies (see Fig. 2.10). We validate the exact and approximate blocking results with the discrete event simulation under the κ-shortest path routing and RF and FF spectrum allocation scenarios, without and with spectrum conversion (SC) capability each (referred to as RF, FF, RF-SC and FF-SC, respectively). For all RF and RF-SC scenarios, in calculating probability of acceptance and approximate EES and SOC BP results, the total exact link-states, non-blocking, and fragmentation link-states are obtained by Eqs. (2.8)–(2.10). For FF and FF-SC scenarios, on the other hand, the number of total, non-blocking, and fragmented link-states are obtained by using the Algorithm 1 for small networks. For medium and large scale networks ($C > 10$), finding the number of non-blocking, blocking and

total exact link-states under the FF policy is computationally challenging, therefore, Eqs. (2.8)–(2.10) are also used for obtaining the number of non-blocking, blocking and total exact link-states, which are used in computing the probability of acceptance of a request on a route.

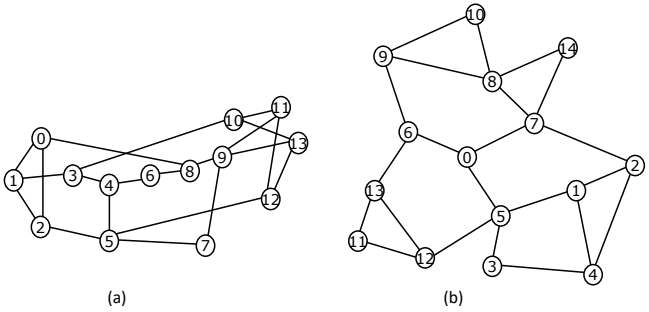

Figure 2.10.: (a) 14-node NSFNET topology, (b) 15-node Atlanta topology.

Blocking results are depicted versus *offered load*, which is defined as $\sum_k \sum_o \frac{\lambda_k^o}{\mu_k}$, with $o \in \mathcal{O}, k = 1, 2, \cdots, K$ connection classes. We assume that the service (holding) times of connection requests between an OD pair are exponentially distributed with mean $1/\mu_k = 1$ unit [20,22], and per-class, per OD pair connection requests arrive according to a Poisson process with uniformly distributed rate λ_k^o= *offered load*$/(|\mathcal{O}| \times K)$. We compute the average BP as $\sum_k \sum_o \lambda_k^o BP_k^o / \sum_k \sum_o \lambda_k^o$, i.e., by ensemble averaging over BP of all OD pairs $o \in \mathcal{O}$ and classes $k = 1, 2, \cdots, K$. All exact and simulation results presented here consider both spectrum contiguity and spectrum continuity constraints for the RF and FF scenarios (i.e., without SC), and only contiguity constraint is considered for the RF-SC and FF-SC scenarios. We generated 10^7 connection requests to simulate each RSA scenario at a given offered load in an EON. We assume that

all links have same capacity C. We consider different combination of bandwidth demands in $\boldsymbol{d} = (3, 4, 6, 10)$ slices, which are equivalent to lightpaths with a guardband on both sides and supporting different bit rates: 10, 40, 100 and 400 gigabit per second using modulation techniques, e.g., QPSK and DP-QPSK, and slice width granularity is 12.5 GHz [60].

Fixed Routing: Figs. 2.11 and 2.12 compare BPs obtained by the exact model and four approximations (EES, SOC, EOV and Uniform) presented in this thesis, and the two known approximations (Kaufman- and Binomial-based, which we implemented according to [22]), under fixed routing ($\kappa_o = 1$) and the RF and FF related scenarios, and validate them by simulation results (Sim.) in a 2-link network with capacity $C = 10$ slices and 3 OD pair routes (shown in Fig. 2.2a). For this small scale ($C = 10$) example, we show approximate BPs separately for RF and FF scenarios (RF-SC and FF-SC scenarios are not shown here as the change in BP results due to SC operation is negligible) using the EES and SOC approximation approaches by computing the number of exact blocking and non-blocking link-states (with a single traversing route) using Eqs. (2.8)–(2.10) under the RF policy, and using Algorithm 1 for the FF policy. Remember that the EOV approach requires all exact link-states, thus it is intractable for medium of large scale EONs. First, we observe that exact BPs match with the simulation results. Thus, in a large scale link or network where the exact solution is intractable, simulation results can be used to verify the approximate solutions. Furthermore, we can see in Fig. 2.11 that the App.EOV and App.EES approaches provide a good approximation for the RF policy without SC. Interestingly, the App.SOC is better than the App.EES or App.EOV for the FF scenario, thus it can be used to estimate approximate blocking probability under the FF policy with and without SC. The Uniform approximation approach overestimates BPs in all cases, as ignores spectrum patterns form by a

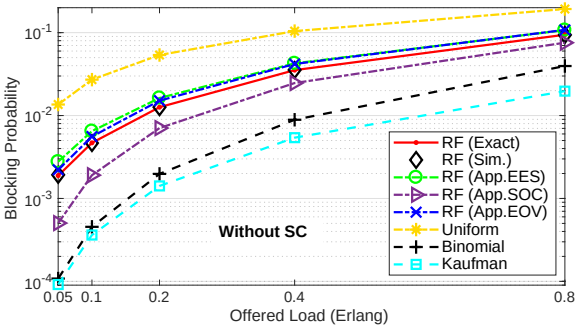

Figure 2.11.: BPs using Exact and various approximation approaches in a 2-link network (Fig. 2.2a) with capacity $C = 10$ slices, and demands $\boldsymbol{d} = (3, 4)$ slices for RF scenario.

given set of demands while computing the probability of acceptance of a request on a route. Thus, to evaluate blocking probability in the following EON cases, we present mainly the EES approximation for the RF and RF-SC, and the SOC approximation for the FF and FF-SC scenarios. The approximations in [22] do not present separate results for the RF and FF policies. In the Kaufman's approach, the link-state distribution is obtained after relaxing the continuity and contiguity constraints. Therefore, the obtained approximate BPs are very low compared to the exact BPs, and they do not match any scenarios. The Binomial approach [21, 22], on the other hand, can be used to obtain approximate BPs under with and without SC separately as it considers the RSA constraints partially. It can be seen that it is a relatively better approach than the Kaufman at higher loads. However, the Binomial approach also underestimates BPs for all traffic loads shown in Fig. 2.11.

To illustrate the accuracy and trend of ours and other approxi-

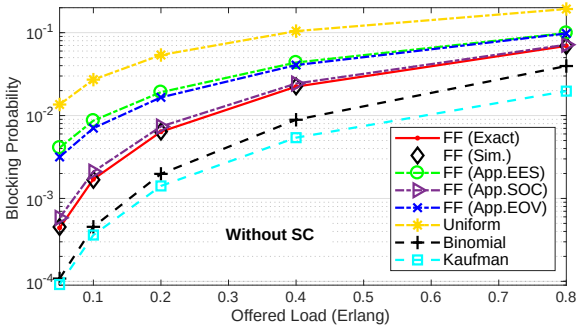

Figure 2.12.: BPs using Exact and various approximation approaches in a 2-link network (Fig. 2.2a) with capacity $C = 10$ slices, and demands $\boldsymbol{d} = (3, 4)$ slices for FF scenario.

mations with higher link capacities and number of hops, Figs. 2.13, 2.14, 2.15, and 2.16 present approximate and verifying simulation results in the same 2-link and in a 10-link linear network topology ($\kappa_o = 1$, and 55 OD pair routes), respectively for different link capacities $C = \{100, 200\}$ slices and demands $\boldsymbol{d}$. We observe that when the link capacity is increased to $C = 100$, the App.EES is closer to simulation results for RF related scenarios in both networks. However, the accuracy of the App.EES reduces for lower loads when the capacity is further increased to $C = 200$, possibly due to round off errors introduced by combinatorial terms with large capacity, and due to inaccuracy involved with the estimation of probability of acceptance terms. Interestingly, the accuracy of the App.SOC for FF scenarios shows relatively good (poor) BP estimation for $C = 200$ ($C = 100$) slices in the 10-link networks at lower loads. In fact, the App.SOC overestimates BPs at lower loads with increasing capacities and hops. The reason is that although the SOC approximation

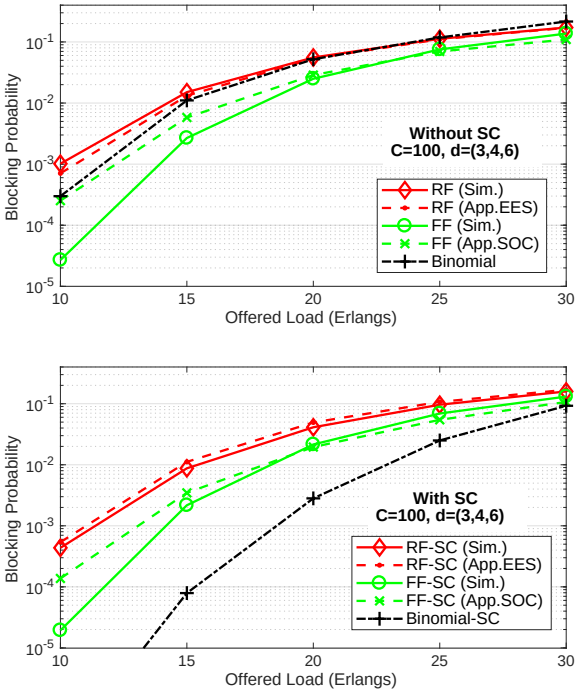

Figure 2.13.: Approximate and simulation BP results in a 2-link network with capacity $C = 100$ slices are shown for RF and FF related scenarios.

assigns higher probability of acceptance to lower occupancy states ($x < \bar{x}$), it might be underestimating the load correlation factor in probability of acceptance at lower loads. The effect of larger number of hops on increasing approximation inaccuracy is also visible at lower loads due to the independence link assumption, but its effect is not as strong as the capacity factor for the RF and RF-SC scenarios. Nevertheless, the accuracy of the App.EES and App.SOC

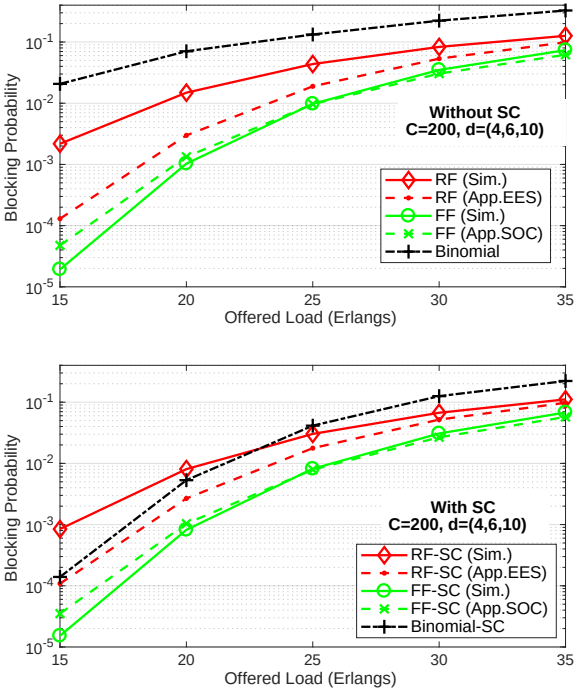

Figure 2.14.: Approximate and simulation BP results in a 2-link network with capacity $C = 200$ slices are shown for RF and FF related scenarios.

approaches increases with increasing loads. Notice that the Kaufman approach is not shown, since it obtains very low BPs. The Uniform approach without SC is intractable for larger hop networks, and it overestimates BPs like the Binomial without SC. The Binomial-SC approach in the 10-link network is also not shown due to very low BPs. Furthermore, we see the same trend as observed in [22] that the Binomial's approach without SC can be used to estimate BPs under

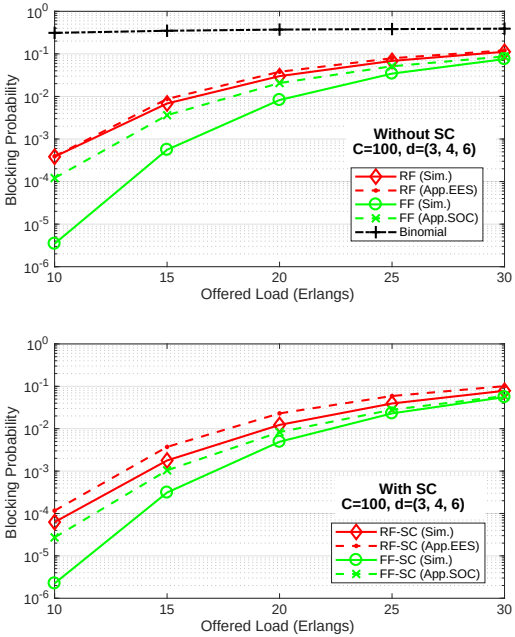

Figure 2.15.: Approximate and simulation BP results in a 10-link network with capacity $C = 100$ slices are shown for RF and FF related scenarios.

the RF scenario for small networks (e.g., 2 hops) where the link-load correlation is negligible. On the other hand, our reduced load link independence model is quite useful even in networks with larger number of hops per route. The reason is that even assuming the statistical independence link, the state-dependent connection arrival rate on a link is computed through the reduced load approximation which considers the spectrum occupancy of each hop of lightpaths routed over the link.

Alternate Routing: To illustrate the accuracy of the exact and

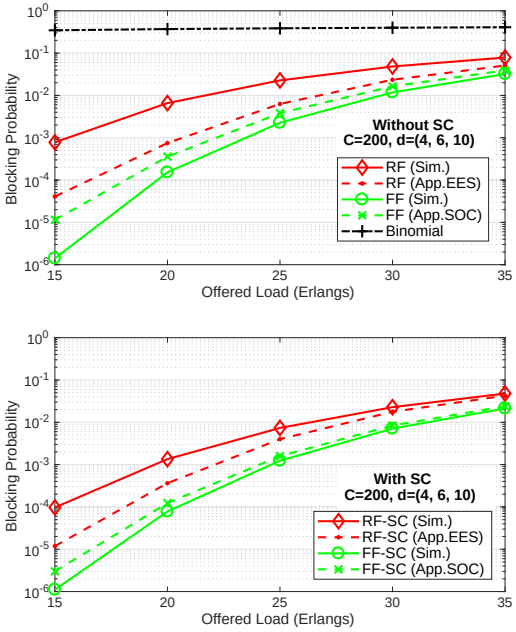

Figure 2.16.: Approximate and simulation BP results in a 10-link network with capacity $C = 200$ slices are shown for RF and FF related scenarios.

approximation models under κ_o paths per OD pair, which is closer to real network scenarios with alternate routing, we plot the exact, approximate, and verifying simulation BP results in Fig. 2.17 for a small 3-node network with 3 OD pairs and 4 routes (see Fig. 2.2b). In Fig. 2.17, traffic load axis is also in log scale to show its non-linear variation, and we observe that exact BPs under all scenarios match the simulation and approximate results. Furthermore, as expected, BPs under the FF-related scenarios are lower than that of the RF scenarios due to the lower spectrum fragmentation [37, 52].

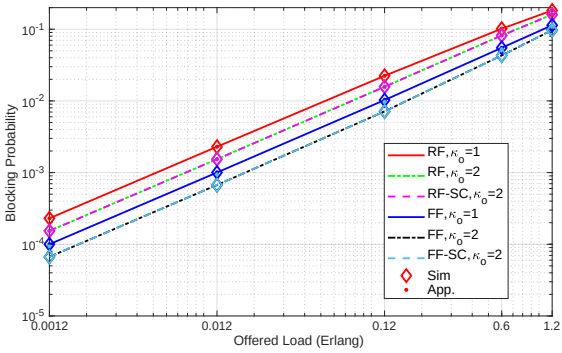

Figure 2.17.: BPs in a 3-node network (shown in Fig. 2.2b) are depicted using the Exact, Sim. and App. methods for $C = 7, \boldsymbol{d} = (3, 4)$.

Notice that when we allow routing on only first shortest path, i.e., $\kappa_o = 1$, all three OD pairs in the 3-node network traverse single-hop paths, thus making SC operation ineffective. When a maximum of 2 shortest paths per OD pair, i.e., $\kappa_o = 2$ is used, we see the blocking reduces under both the RF and FF related scenarios in the 3-node network, even though only one OD pair (node 1 to 3) has two routes. Thus, the spectrum conversion is only utilized for traffic from node 1 to 3 on its second shortest path $1 \rightarrow 2 \rightarrow 3$, and it shows as beneficial as κ-shortest path routing. When the network-link capacity is increased to $C = 100$ slices in Fig. 2.18, the approximation (EES) BP results obtained under the RF related scenarios are still very close to the simulation results. On the other hand, the approximation results (SOC) for the FF related scenarios are overestimated for lower loads, as observed in Figs. 2.13–2.16.

Next, we show the BP approximations for all scenarios in a real network topology. We consider a well known 14-node NSFNET

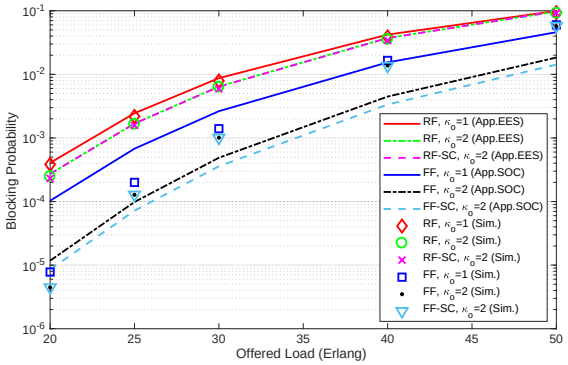

Figure 2.18.: BPs in a 3-node network (shown in Fig. 2.2b) are depicted using Sim. and App. methods for $C = 100, \boldsymbol{d} = (3, 4, 6)$.

topology with 42 unidirectional links and all possible OD pairs ($|\mathcal{O}| = 182$) and $\kappa_o = \{1, 2\}$ routes per OD pair. Figs. 2.19 and 2.20presents the approximate BP results and verifying simulations in the NSFNET under various RF and FF related scenarios, respectively. Unlike the 3-node network system, here we observe that the SC operation is very helpful in reducing blocking, and BP reduces considerably under the RF-SC scenarios, as compared to the RF, i.e., without SC. The reason is that the RF scenario tries to assign random continuous and contiguous free slices to a new request (which results in higher spectrum fragmentation), and with SC, the continuity constraint is relaxed when it does not find the required aligned free consecutive slices over a route. However, the FF-SC operation still offers the lowest BPs as compare to other scenarios. Allowing two paths per OD pair ($\kappa_o = 2$) for the RSA outperforms the shortest path scenarios, and decreases blocking by one or more order of magnitudes at lower loads.

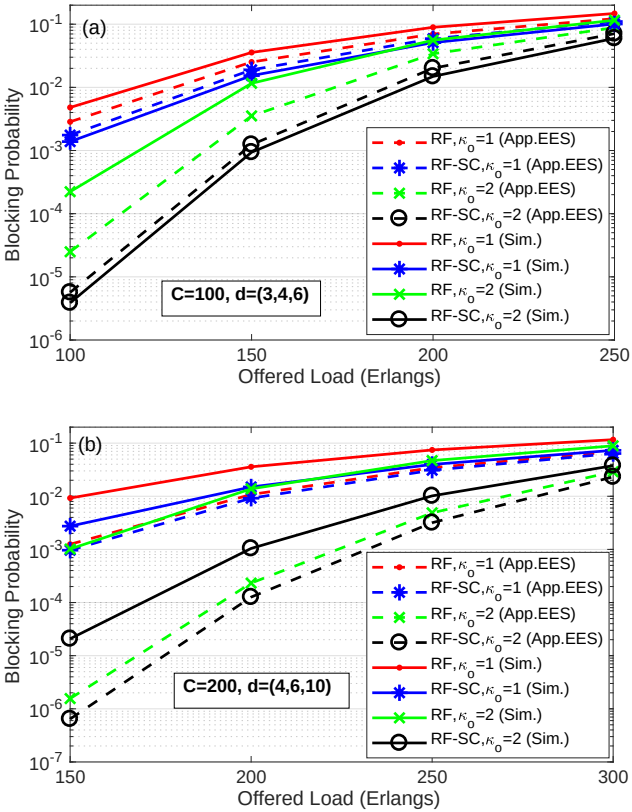

Figure 2.19.: Approximate and simulation BP results in a NSFNET are depicted for RF related scenarios.

The EES (SOC) approximation used to obtaining BPs under the RF (FF) related scenarios are very close to the simulation results for $\kappa_o = 1$ in Fig. 2.19. For $\kappa_o = 2$, the approximate BPs of the scenarios with SC operation, e.g., RF-SC, are either underestimated

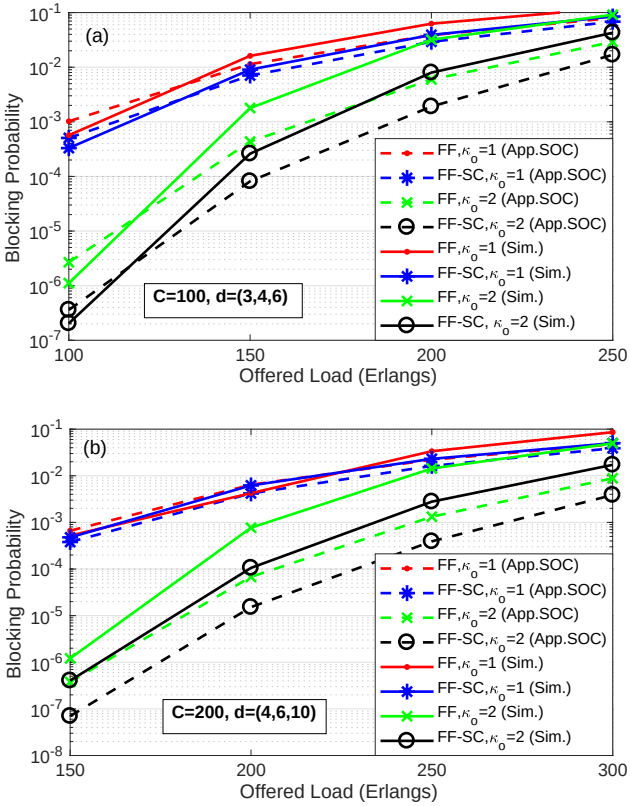

Figure 2.20.: Approximate and simulation BP results in a NSFNET are depicted for FF related scenarios.

or overestimated, but they are also close to the simulation results. However, for scenarios without SC, e.g., RF, the approximate BP results are underestimated for $\kappa_o = 2$, and they deviate by a bigger margin at lower loads and for higher link capacity $C = 200$. There could be various reasons for such behavior. First, we assume that

connection requests arrive on each link according to a Poisson process. However, an OD pair request is accepted on alternate routes ($\kappa_o \geqslant 2$) when previous shorter routes do not have sufficient free slices. Thus, the traffic arrival on alternate routes and therefore on a link is not a Poisson process. Second, the approximate probability of acceptance obtained in this thesis does not consider the load-correlation factor between links [9]. Third, at lower loads all possible link-states in a large capacity link could not be sufficiently visited even simulating with a large number of events, so simulation results might also not be very accurate for $C \geqslant 200$ cases. Forth, the EES ans SOC approximations involve combinatorial terms, which introduce round off errors in probability of acceptance computation in a large capacity network scenario. However, for most cases we observe that approximate BPs are very close to the simulation results, thus making them acceptable. We observe the similar trend in a 15-node Atlanta topology [58].

As both RF and FF policies have some advantages and disadvantages, e.g., the FF policy is preferable for lower blocking, whereas the RF policy is suitable for load balancing, security in EONs [64, 65]. Thus, both policies can be made useful for deployment of new services. In summary, we can say that the EES (SOC) approach can be used to estimate BP with or without SC under the RF (FF) policy even in a large scale networks. More importantly, allowing RSA on κ-shortest path is beneficial than utilizing single shortest path routing. However, the accuracy of the approximations deviates with consideration of more number of paths per OD pair. The approximation approaches could be further improved by utilizing a more accurate probability of acceptance of a request in EONs without SC, and considering the link correlation model used in WDM networks [9], which is relatively complex compared to the link independence model but essential to analyze the effect of correlation of loads among links under different RSA policies in a network. At

the same time, the scalability of load and link correlation models could be a major issue that needs to be worked on in the future. To this end, we suggest that the effect of correlated links can be looked into by considering an occupancy correlation factor on adjacent links while computing the connection arrival rates. Furthermore, a less complex approximation for a partial SC operation (where only some nodes have SC) can be worked on to obtain BPs which are closer to without SC scenarios.

Complexity Analysis: The time complexity of the exact Markov model varies based on the applied RSA policy, and it grows exponentially with link capacity C, number of routes in $\mathcal{O}$, maximum number of hops on a route H, and demand classes K. Thus, the overall complexity is non-polynomial in time, and it requires $\sim O(2^{|\mathcal{O}|\kappa_o CHK})$ computation. For the evaluated 3-node network scenario, this can translate to several hours to compute the exact blocking probability on a commodity computer. Nevertheless, the exact blocking analysis is useful for a small scale network, and the exact link-state information corresponding to fragmented and non-fragmented states could be used to computing approximate probability of acceptance of connection requests. The time complexity of the reduced state model mainly depends on the computation of link-state occupancy distribution $\boldsymbol{\pi}$, connection setup rates $\boldsymbol{\alpha}$, departure rates $\boldsymbol{\gamma}$ and blocking probabilities of all OD pairs BP_k^o. However, the overall complexity is dominated by the computation of class- and state-dependent connection setup rates $\boldsymbol{\alpha}$. The reason is that in each fixed-point iteration, $\boldsymbol{\alpha}$ needs the knowledge of the probability of acceptance terms for all combination of occupancy states on all (say, H-hop) routes. Thus, for all links $|\mathcal{J}|$, classes K and routes per link R, it generally requires non-polynomial $O(|\mathcal{J}|KRHC^H)$ computation [55], using the Uniform and the EOV approximation approaches. As shown in Section 2.6, we overcome this problem by simplifying the sum of product computation in Eq. (2.28), using a product-form approximation of

the probability of acceptance in Eqs. (2.19)–(2.22), to product of sum terms that require only $O(|\mathcal{J}|KRC^2H^3)$ computation in the EES and SOC approaches without SC (and $O(|\mathcal{J}|KRC^2H^2)$ with SC), which is polynomial. The state probabilities on all links requires $O(|\mathcal{J}|C^2)$ computation using the LSQR method, and departure rates and the computation of link-states need the information of the possible set of connections per class $\boldsymbol{n}$, which requires $O(K^2C)$ computation. The BP_k^o reuses probability terms in $\boldsymbol{\alpha}$, thus requires $O(CK|\mathcal{O}|\kappa_o)$ computation. Therefore, the overall complexity of each iteration is $\sim O(|\mathcal{J}|KRC^2H^3 + CK|\mathcal{O}|\kappa_o)$, as compared to the Binomial approach with overall complexity $\sim O(C^2 + CK)$. Thus, the EES and SOC approximations are good to obtain more accurate BPs, but at the cost of higher complexity. Note that the convergence of the fixed-point iteration algorithm is still unproven [55]. However, for all scenarios and topologies presented here, the algorithm takes around 5 to 20 seconds per fixed-point iteration, and converges after 7 to 30 iterations on a commodity computer (Intel single-core i7 3.20 GHz processor with 8 GB RAM).

2.8. Summary

In this chapter, we proposed the first exact Markov model for modeling the routing and spectrum allocation policies in EONs, and subsequently the related methods to reducing the exact model into a reduced occupancy model to computing approximate blocking. More in detail, we presented load-independent and load-dependent approximations to compute the probability of acceptance of a request in EONs with and without spectrum conversion, considering bandwidth demands, contiguity constraint and continuity constraint. These approximations use the information of the number of non-blocking and blocking exact states corresponding to an occupancy state, which we derive for a random-fit assignment policy using the

inclusion-exclusion principle. Additionally, the exact and approximate BPs are presented for cases with and without spectrum conversion under κ-shortest path routing, and a random-fit and a first-fit spectrum allocation policies. The numerical results obtained show that allowing routing and spectrum allocation on alternate routes, and spectrum conversion are beneficial in terms of blocking probability reduction. The exact blocking analysis is accurate, albeit limited to a very small scale EONs, due to complexity. On the other hand, approximate solutions have been shown accurate in a broader range of scenarios. It was shown in fact that the accuracy of the approximation methods proposed depends on the various factors, such as the RSA policies, link capacity, traffic loads, hop-length and topology. The next steps in this line of work include major challenges that have not been solved yet, and are analytically rather complex, most notably the inter-dependency of network link loads.

3

Analysis of Resource Reallocation in Elastic Optical Networks

3.1. Introduction

Spectrum fragmentation in EONs occurs due to dynamic setup and termination of heterogeneous demands over continuous and contiguous spectrum paths, which leave the gap between occupied spectrum slices. This generally lowers the spectrum utilization and increases connection blocking in EONs. To address the issue of fragmentation, many prior works have focused on developing efficient RSA schemes or some kind of fragmentation-aware strategy in order to reduce spectrum fragmentation and connection blocking in EONs [12, 25, 60, 66]. However, as network traffic varies with time, defragmentation as a connection reconfiguration scheme is desirable, since spectrum allocation might not be optimal. Defragmentation (DF) in optical networks is performed by reallocating spectrum to some of the existing connections such that a larger free block of spectrum is created to accept new demands. DF in general is an NP-hard problem in an EON, and it can be triggered reactively or proactively, and connections can be reconfigured in a sequential or parallel manner, as shown by an integer linear programming approach in [32].

This chapter first proposes an analytical framework to model a DF

process in a generalized form that allows both proactive and reactive triggering mechanism for connection reconfiguration an elastic optical link (EOL) using a CTMC model. Following which, we use the same exact and approximate CTMC models presented in the previous chapter with additional DF states to analyze the effect of defragmentation on the overall blocking in EONs under the fixed routing and the random-fit (RF) or the first-fit (FF) spectrum allocation schemes. In various DF models, the DF states are introduced to reconfigure some of the connections and accept a new request which could not be accepted otherwise in the system (i.e., without DF). Again, a "state" of a Markov model could represent either an exact state or a microstate depending on the applied exact or reduced state model, respectively. Notably, the presented analytical (exact and approximate) results are very interesting as they find individual blocking parts – *resource blocking*, which occurs due to the unavailability of free spectrum resources, and *fragmentation blocking*, which occurs due to the fragmentation of spectrum occupancy. Additionally, we show a fraction of overall blocking that can not be eliminated in a mesh topology under a given RSA policy even with an *optimal* DF scheme.

3.2. Supporting Publications

- S. K. Singh and A. Jukan, "Computing Blocking Probabilities in Elastic Optical Networks with Spectrum Defragmentation," *IEEE International conference on Computer Communications (INFOCOM)*, Paris, April 2019.

- S. K. Singh, W. Bziuk, and A. Jukan, "Analytical Performance Modeling of Spectrum Defragmentation in Elastic Optical Link Networks," *Optical Switching and Networking*, vol. 24, pp 25-38, 2017.

- S. K. Singh, W. Bziuk, and A. Jukan, "A Combined Optical Spectrum Scrambling and Defragmentation in Multi-Core Fiber Networks," *IEEE International Conference on Communications (ICC)*, Paris, France, May 2017.
- S. K. Singh, W. Bziuk and A. Jukan, "Balancing Data Security and Blocking Performance with Spectrum Randomization in Optical Networks," in *IEEE Global Communications Conference (GLOBECOM),* Washington, USA, December 2016.
- S. K. Singh, W. Bziuk and A. Jukan, "Balancing Security and Blocking Performance with Reconfiguration of the Elastic Optical Spectrum," in *International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO),* Croatia, May 2016.
- S. K. Singh, W. Bziuk and A. Jukan, "Defragmentation-as-a-Service (DaaS): How beneficial is it?," in *Optical Fiber Communication (OFC) Conference,* Anaheim, CA, USA, March 2016.

3.3. A Framework for Modeling of an Exact DF Process in an EOL

Spectrum defragmentation is a process where resources are reallocated to reconfigure some or all connections such that a large continuous and contiguous free block of spectrum is created to accept future requests. For the spectrum defragmentation, the most favored scheme is "move-to-vacancy", which shifts a connection only to a sufficient large block of free spectrum slices. There are two major methods that have been proposed for the reconfiguration: sequential and parallel defragmentation [32]. An example for parallel defragmentation is shown in Fig. 3.1(a). When we apply DF on the fragmented link, connections c_2 and c_3 are moved in parallel to free

spectrum slices, whereas connection c_1 is not affected. Note that in this example two connections can be reconfigured in parallel in t_d time unit. The sequential defragmentation is illustrated in Fig. 3.1(b). Here, at first connection c_3 is moved to a free slice and in the next step connection c_2 is moved to available slices freed in the first step. Thus, in this example $\eta = 2$ steps are required with a total disruption time of $2t_d$. The number of required steps, η, depends on the spectral connection pattern (which we will associate later to a system state) and the used combination of sequential and parallel reconfigurations determined by an optimization algorithm [32]. To allow a generalized modeling, we use a state depending exponentially distributed reconfiguration time $T_{RT}(S_{d_\nu})$, which has a mean value $\eta_\nu t_d$, where $t_d = 1/\mu_d$ and η_ν is the number of reconfiguration steps for a DF state S_{d_ν}. This enables us to model both sequential as well as parallel reconfiguration schemes. As defragmentation optimization is not the target in this section, we assume a simple method, where η_i is equal to the number of contiguous free spectrum blocks in the range between the lowest spectrum slice (i.e., index=1) and the highest occupied spectral slice. For example, in Fig. 3.1(a) only $\eta_\nu = 1$ step is required because before DF there is a single block of contiguous free slices in between lowest slice 1 and highest occupied slice 7, whereas in Fig. 3.1(b) $\eta_\nu = 2$ reconfiguration steps are required due to two blocks of contiguous free slices in spectral range [1, 6] in an occupancy state before DF operation.

Now, let us present an example of a generalized DF framework that allows both proactive and reactive triggering mechanism for connection reconfiguration an EOL, where there is only one origin-destination pair ($o = 1$) which is omitted from the parameters used in this section. Fig. 3.2 illustrates three different DF models under dynamic connection set up and termination scenario with 3 (shown in green color) and 4 (shown in yellow color) consecutive slices arriving demands in a 7 slices fiber-link. Here, let us assume that a

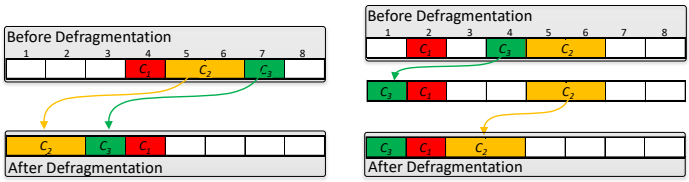

Figure 3.1.: Reconfigutation of connections with demands $\boldsymbol{d} = (1, 2)$ slices during DF period in an EOL [37].

demand with 3 slices bandwidth arrives at time $t_0 = 0$ in an empty system state (not shown) and the system randomly allocates three consecutive slices (3–5) to this request, as shown in a state S_4. Note that the state S_4 can not accept a 3 slices or a 4 slices request due to the unavailability of required consecutive free slices (i.e., fragmentation of spectrum). To consolidate free slices, therefore, a DF event can be triggered either proactively or reactively. In Fig. 3.2(a), system monitors the fragmentation proactively, and triggers a DF operation at time $t_1 = T_P > 0$, where T_P is the fragmentation detection time. After reconfiguring the spectral resources assigned to the connection in a DF state S_{d_1} in the time interval of $T_{RT} > 0$, the system returns to a non-fragmented state S_2 at time $t_2 = T_P + T_{RT}$. In Fig. 3.2(b), on the other hand, DF is reactively triggered when a new connection request with demand 3 (or 4) slices arrives at time T_{E_1} (or T_{E_2}). Although the DF is started, this connection request will not be accepted in the state S_4 and is blocked. In contrast, in the delayed (DL) method, the set-up of the triggering request will be delayed and not blocked. This is shown in Fig. 3.2(c) for the hybrid Pro-Re-DL-DF model, when a new 3 (or 4) slices request arrives in the state S_4 and such a triggering request is delayed for the time duration of the DF period i.e., T_{RT}. When the reconfiguration of an existing connection is completed in the DF state (S_{d_2}) then it

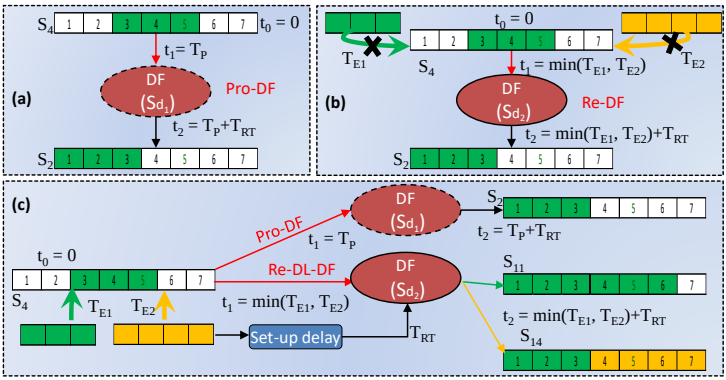

Figure 3.2.: Link state transitions with demands $\boldsymbol{d} = (3, 4)$ slices in (a) Pro-DF model, (b) Re-DF model and (c) Pro-Re-DL-DF model [37].

immediately allocates next available 3 (or 4) free slices to the request and transits to a state S_{11} (or S_{14}). On the other hand, when the system decides to perform DF proactively before one of the two requests arrives i.e., $T_P < min(T_{E_1}, T_{E_2})$, then proactive DF takes place in DF state S_{d_1}, and the system returns to the state S_2. For all systems shown in Fig. 3.2, it should be noted that further requests are blocked if they arrive during the reconfiguration time, i.e., if the system is in a DF state.

Assumptions of the DF Link Models

The following assumptions allow us to statistically analyze the performance of the system.

(i) Requests' inter-arrival times, connections' service times and reconfiguration times are independent and exponentially distributed.

(ii) The reactive DF process is triggered, when a request enters into a fragmented system state where it could not be served normally.

(iii) If the system enters a fragmented state, the proactive DF process is triggered after an exponentially distributed fragmentation detection time.

(iv) Services of existing connections are interrupted during reconfiguration time and are resumed when DF is completed.

(v) All incoming requests are blocked during the reconfiguration time, which is termed as defragmentation blocking.

(vi) The request which triggers the reactive DF process is put on hold (delayed) in the Pro-Re-DL-DF model, and starts its service immediately after the defragmentation.

As usual, assumption (i) allows to model the system using the Markov chain approach. Furthermore, the reconfiguration time is assumed to be exponentially distributed and can depend on the system state. This property has been chosen, because it allows to model the spectrum reconfiguration independent of a specific algorithm, but depending on the actual spectral usage. The assumption of an exponentially distributed fragmentation detection time (iii) not only allows a theoretical modeling, but also takes into account that a practical proactive triggering method will not use a high priority deterministic state observation method, thus the detection time can be assumed to be statistically distributed. A drawback of the DF process is the fact that traffic interruption (iv) of some but not all existing connections happens during reconfiguration in the real system, however reconfiguration can be performed in much lower time ($\sim$ microseconds) [46, 67]. Another drawback of the DF process is that all incoming requests for all types of connections are blocked during the reconfiguration process, i.e. assumption (v). This fact

is partially relaxed for the Pro-Re-DL-DF model, which accepts the incoming triggering request (vi). This assumption can be realized in a real system by simply delaying the connection setup procedure in the control plane until the DF has finished. As we analyze a single-hop network in this section, which serves as a preliminary step towards analyzing multi-hop networks by assuming statistical independence of the link state distributions, only the spectrum contiguity constraint is considered. The notation and parameters used in the models are given in Table 3.1.

3.3.1. State-Space Generation and Identifying Blocking States

To model the normal behavior of a link system (i.e., without DF) using the CTMC model, we adopt the same technique presented in the previous chapter to generate and represent all regular link-states $S_i, i = 1, \cdots, |\Omega_S|$. Additionally, to model the reconfiguration (DF) process, we introduce special DF states $S_{d_\nu}, \nu = 1, \cdots, |\Omega_D|$. Only to simplify the notation for a joint modeling of all models, we generate a DF state for every fragmented state. We can also merge two or more DF states having same reconfiguration rate if the transitions leading from these DF states result into same occupancy pattern (exact state), see [43]. This can reduce the complexity of the CTMC, which grows exponentially with the number of states. However, here we avoid this because one-to-one mapping from a normal state (S_i) to a DF state (S_{d_ν}) allows us to model state dependent reconfiguration rate ($\mu_d(S_{d_\nu})$), and the same model can be used to analyze proactive and reactive DF.

We assume that both inter-arrival times T_{E_k} and connection holding times T_{H_k} of class-k requests are independent and exponentially distributed with average rates λ_k and μ_k, respectively. To identify the blocking states, their dependence on the connection classes, and the DF triggering events in all models, in Fig. 3.3 let us consider of an EOL with capacity $C = 7$ slices, with two different classes (class 1

Table 3.1.: Notations and the parameters used in the DF models [37].

Notation	Description
C	Total number of spectrum slices (or capacity units)
K	Number of classes of connections. Note that a class $k = 1, 2, \cdots, K$
$\lambda_k(\mu_k)$	Arrival (service) rate of class k connections
λ_P	Arrival rate of a proactive DF request (a class-0 request i.e, $\lambda_P \equiv \lambda_0$)
$\boldsymbol{d}$	$\equiv (d_1, d_2, \ldots, d_K)$, where d_k: bandwidth (in slices) of class k
$\boldsymbol{n}$	$\equiv (n_1, n_2, \ldots, n_K)$, where n_k is the number of class k connections
S_i	Occupancy state for normal operation $i = 1, ..., \lvert\Omega_S\rvert$
S_{d_ν}	Defragmentation (DF) state $\nu = 1, ..., \lvert\Omega_D\rvert$
$\mu_d(S_{d_\nu})$	Mean defragmentation rate of DF state S_{d_ν}
$\overleftarrow{S_i}$	State with shifted spectrum pattern of S_i due to DF, $i = 1, ..., \lvert\Omega_S\rvert$
$IS(i)$	$\nu = IS(i)$ index of DF-state S_{d_ν} which is bijective mapped to S_i
$n_k(S_i)$	Number of class k connections in state S_i
$\boldsymbol{n}(S_i)$	$\equiv (n_1(S_i), \ldots, n_K(S_i))$, realization of $\boldsymbol{n}$ in state S_i
$\Gamma^{k+}_{S_i}$	A set of regular states after a class k request arrives in state S_i
$\Gamma^{k-}_{S_i}$	A set of regular states after a class k request departs from state S_i
$f_l(S_i)$	Largest size of contiguous free slices in a state S_i
$\mathbb{FI}(k)$	Set of fragmentation states for class k requests
$\mathbb{FC}(S_i)$	Set of classes for which S_i is the fragmentation state
$\mathbb{TI}^M(S_i)$	Set of classes for which S_i triggers the defragmentation method M
$\mathbb{RB}(k)$	Set of resource blocking states for class k requests
$\lvert\Omega_S\rvert$	Number of link-states without defragmentation in a given SA policy
$\lvert\Omega_D\rvert$	Number of defragmentation (DF) states in a given SA policy

and 2) of requests, and demands of 3 and 4 consecutive slices for all models. The current state S_4 serves a class 1 request with demand $d_1 = 3$ slices. In this example, S_4 is a fragmented state, wherein any arriving request of both classes will be blocked. We identify class k fragmented states which can not allocate its largest block $f_l(S_i)$ of l consecutive free slices to an incoming class k request of demand

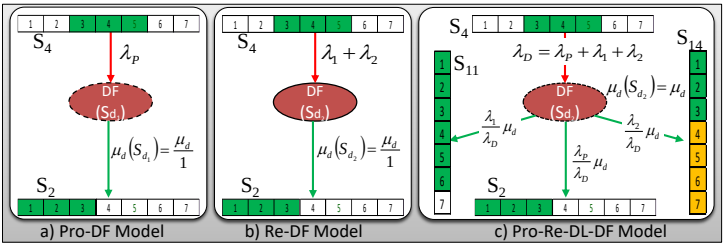

Figure 3.3.: State transitions from and into a DF state with demands $\boldsymbol{d} = (3, 4)$ slices. a) Proactive DF model, b) Reactive DF model, and c) Proactive-Reactive-Delay DF model [37].

$d_k > l$, even though it contains equal or more than d_k number of free slices, given as

$$\mathbb{FI}(k) = \{S_i | d_k > f_l(S_i) \,\&\&\, d_k \leq C - \sum_j d_j n_j(S_i), i = 1, \cdots, |\Omega_S|\}. \tag{3.1}$$

Taking into account all classes, we have $\mathbb{FC} = \bigcup_k \mathbb{FI}(k)$. Thus, we can define the set of classes for which state $S_i \in \mathbb{FI}$ is the fragmentation state as

$$\mathbb{FC}(S_i) = \{k \mid S_i \in \mathbb{FI}(k), k = 1, ..., K\}, i = 1, \cdots, |\Omega_S|. \tag{3.2}$$

Resource blocking states, on the other hand, do not have sufficient free slices to satisfy an incoming class k request, i.e.,

$$\mathbb{RB}(k) = \{S_i | d_k > C - \sum_j d_j n_j(S_i), i = 1, \cdots, |\Omega_S|\}. \tag{3.3}$$

In Fig. 3.3(a), DF is proactively triggered by the Pro-DF scheme when it detects the current state as a fragmentation state, e.g. the system moves from state S_4 to a DF state S_{d_1}. This happens after a random fragmentation detection time interval $T_P \sim \exp(\lambda_P)$, which we assume to be independent and exponentially distributed with average rate λ_P. This is a state dependent arrival, which means that

Pro-DF is triggered only from a set of states that are fragmented states, i.e. $S_i \in \mathbb{FI}$. As discussed above, we assume that the DF reconfiguration times are also exponentially distributed with state depending rate $\mu_d(S_{d_\nu}) = \mu_d/\eta_\nu$, where η_ν is the number of reconfiguration steps required for the defragmentation in a DF state S_{d_ν}.

In the Re-DF scheme (see Fig. 3.3(b)), an arrival of a class 1 or class 2 request will be blocked in the fragmented state S_4 as well, but also triggers the system to transit from S_4 to a DF state S_{d_2}. This triggering event will happen after the first request arrives, which means after the time interval $min\{T_{E_1}, T_{E_2}\}$. As a well known fact, the minimum of the two independent and exponentially distributed random times results in the combined arrival rate $\lambda_1 + \lambda_2$.

Unlike the Pro-DF and Re-DF models, Pro-Re-DL-DF model (see Fig. 3.3(c) and also Fig. 3.2(c)) accepts requests which would have been blocked otherwise due to fragmentation. As a special feature, this model allows to combine proactive and reactive triggered DF using one combined DF state. Therefore, the transition from the state S_4 to S_{d_2} happens after a random time interval $min\{T_P, T_{E_1}, T_{E_2}\}$, which results in the combined arrival rate $\lambda_D = \lambda_P + \lambda_1 + \lambda_2$. To simplify notation, the triggering of Pro-DF can be viewed as an arrival of a *class-0* request with rate $\lambda_0 = \lambda_P$ and demand $d_0 = 0$ slice. In the example of Fig. 3.3(c) the system reconfiguration occurs in the DF state S_{d_2}, and depending on the cause of trigger (pro-DF or Re-DF) the transition from S_{d_2} to different target states takes place. After defragmentation, system allocates the spectrum slices to a request R_k and returns to a state S_{11} or S_{14} if the request R_k belongs to a class 1 or class 2, or to S_2 for the virtual class-0 request, respectively. The transition rate from S_{d_2} to $S_i, i = \{2, 11, 14\}$ can be calculated as the DF rate $\mu_d(S_{d\nu})$, multiplied by the probability of the arrival of a class-k request in the state S_{d_2}, i.e., $\frac{\lambda_k}{\lambda_D}$, where $k \in \{0, 1, 2\}$ depends on the state.

For a unified modeling, M is used to identify the different DF

models, i.e., $M = I$, $M = II$ and $M = III$ denote Pro-DF, Re-DF and Pro-Re-DL-DF models, respectively. This allows to define the set of triggering classes per state as follows.

$$\mathbb{TI}^M(S_i) = \begin{cases} \{0\} & \text{if } M = I \text{ and } S_i \in \mathbb{FI} \\ \mathbb{FC}(S_i) & \text{if } M = II \text{ and } S_i \in \mathbb{FI} \\ \mathbb{FC}(S_i) \cup \{0\} & \text{if } M = III \text{ and } S_i \in \mathbb{FI} \\ \phi & \text{otherwise} \end{cases} \tag{3.4}$$

As an example, in Fig. 3.3(c) we have $\mathbb{TI}^{III}(S_4) = \{0, 1, 2\}$. Similar to the previous chapter, the transitions from a normal state S_i to another state S_j or a DF state S_{d_ν} and vice-versa, can be easily identified to write the GBE of all normal states S_i, $i = \{1, \cdots, |\Omega_S|\}$ and DF states S_{d_ν}, $\nu = \{1, 2, \cdots, |\Omega_D|\}$ [37]. Under the stationary condition, the state probabilities $\boldsymbol{\pi} = [\pi(S_1), \pi(S_2), \cdots, \pi(S_{|\Omega_S|}), \pi(S_{d_1}), ..., \pi(S_{d_{|\Omega_D|}})]$ can be calculated by solving $\boldsymbol{\pi Q} = 0$ subject to $\sum_i \pi(S_i) + \sum_\nu \pi(S_{d_\nu}) = 1$, where $N = |\Omega_S| + |\Omega_D|$ is the total number of states, and $\boldsymbol{Q}$ is the transition rate matrix.

Calculation of Performance Measures

Let us now calculate the performance measures, such as set-up delay time in our Pro-Re-DL-DF model (III), and blocking probabilities in all models. In model III, a class-k request arriving into a fragmented state S_i (i.e., $k \in FB(S_i)$) is delayed until reconfiguration is performed, and then accepted immediately after the completion of reconfiguration time, and therefore the average set-up delay time is given by

$$\tau_d = \sum_{\nu=1}^{|\Omega_D|} \left[\frac{1}{\mu_d(S_{d_\nu})} \times \frac{\pi(S_{d_\nu})}{\sum_{\nu=1}^{|\Omega_D|} \pi(S_{d_\nu})} \right]. \tag{3.5}$$

The blocking per class k due to the non-availability of resources, i.e., resource blocking probability (RBP), can be calculated for class

k by summing over the states which are resource blocking states for the class k requests, i.e. $S_i \in \mathbb{RB}(k)$.

$$RBP(k) = \sum_{S_i \in \mathbb{RB}(k)} \pi(S_i), k = 1, \cdots, K \tag{3.6}$$

Similarly, the blocking per class k due to the fragmentation, i.e., fragmentation blocking probability (FBP), can be calculated for class k by summing over the states which are fragmentation states for the class k requests, i.e. $S_i \in \mathbb{FI}(k)$.

$$FBP(k) = \sum_{S_i \in \mathbb{FI}(k)} \pi(S_i), k = 1, \cdots, K \tag{3.7}$$

Since DF states are blocking states for all classes of requests, therefore the defragmentation blocking probability (BBP) is given as

$$DBP = \sum_{\nu=1}^{|\Omega_D|} \pi(S_{d_\nu}) \tag{3.8}$$

The overall blocking for the combined Pro-Re-DL-DF model (III) can be calculated by taking into account the resource and the defragmentation blocking, which is given as

$$BP^{III} = DBP + \frac{\sum_{k=1}^{K} \lambda_k RBP(k)}{\sum_{k=1}^{K} \lambda_k} \tag{3.9}$$

The overall blocking for Pro-DF model (I) and Re-DF model (II) can be calculated by taking account into all three types of blocking. These two models immediately blocks arriving requests in fragmentation states like a regular system. The overall blocking for these two models is given as below.

$$BP^{I} = BP^{II} = DBP + \frac{\sum_{k=1}^{K} \lambda_k \left(RBP(k) + FBP(k)\right)}{\sum_{k=1}^{K} \lambda_k} \tag{3.10}$$

In the regular system without defragmentation, in additional to the blocking due to non-availability of resources (Eq. (3.6)), blocking

also occurs due to the fragmentation (Eq. (3.7)). The total blocking is thus given by

$$BP^{Regular} = \frac{\sum_{k=1}^{K} \lambda_k (RBP(k) + FBP(k))}{\sum_{k=1}^{K} \lambda_k}. \tag{3.11}$$

3.3.2. Performance Evaluation

Table 3.2.: Blocking probability (in %) is given for different models under FF and RF spectrum allocation methods for small scale EOL with capacity C=20 slices, and demands $\boldsymbol{d} = (4, 6, 8)$ slices [37]. Confidence interval of the Sim. results is not shown due to their small values.

			Link-Load (Erlangs)					
			2		6		10	
Models	μ_d	SA-policy	Exact	Sim.	Exact	Sim.	Exact	Sim.
Regular	-	FF	1.14	1.14	8.74	8.73	18.55	18.56
		RF	2.96	2.96	13.64	13.64	24.40	24.42
Pro-DF	1	FF	1.20	1.20	9.99	9.99	22.05	22.05
		RF	3.53	3.53	19.06	19.05	35.09	35.11
	100	FF	1.12	1.12	8.54	8.55	18.15	18.13
		RF	2.83	2.84	12.72	12.71	22.64	22.63
Re-DF	1	FF	1.20	1.20	10.15	10.11	22.47	22.49
		RF	3.72	3.72	20.59	20.56	37.33	37.31
	100	FF	1.12	1.11	8.50	8.49	18.05	18.05
		RF	2.79	2.78	12.35	12.35	22.04	22.03
Pro-Re-DL-DF	1	FF	1.04	1.04	10.02	10.00	23.20	23.20
		RF	2.46	2.46	19.81	19.82	37.65	37.68
	100	FF	0.89	0.89	7.30	7.30	16.39	16.39
		RF	0.91	0.90	7.42	7.42	16.60	16.59

Let us now evaluate the performance of all DF models in terms of blocking probability (BP) and set-up delay time in our hybrid DF model (III), and we compare them against the regular system

without defragmentation. We present the analytical results for a small scale EOL having capacity C = 20 spectrum slices and demands $\boldsymbol{d} = (4, 6, 8)$ slices, and validate the model with the Monte Carlo (MC) simulation results using 10 million connection request arrivals. For a large scale scenario, we consider an EOL having capacity C = 100 spectrum slices and demands $\boldsymbol{d} = (5, 10, 15)$ slices, and present only the MC simulation results, since the computation complexity of analytical models increases exponentially with the capacity of an EOL. All the results presented in this section are for an EOL, i.e., for a single-hop network. Mean fragmentation detection time ($T_P = 1/\lambda_P$) is {0.01, 0.001}, and Reconfiguration rate (μ_d) is {1, 100, 1000}. Furthermore, we present all the results under the assumption of spectrum contiguity, i.e., only contiguous slices can be allocated to the requests. The requests are generated according to Poisson process with arrival rates λ_k requests per unit time, and their service duration is exponentially distributed with mean $1/\mu_k = 1$ time unit for all classes of requests. Arrival rates are uniformly distributed i.e., $\lambda_k = \frac{\lambda}{K}$, where λ is the total arrival rate. Load on the link (i.e., Link-Load) is calculated as $\sum_{k=1}^{K} \frac{\lambda_k d_k}{\mu_k}$. For a given load and departure rate $\mu_k = 1$, the uniform connection arrival rate per class λ_k is given by Link-Load$/\sum_{\forall k} d_k$.

The exact performance of a small scale EOL with C = 20 slices, and verifying simulation results in terms of overall blocking probability are shown in Table 3.2 for the regular system and all three DF models under FF and RF spectrum allocation policies. Here, we compare the BP of the regular system against our DF models by considering different DF rates $\mu_d = \{1, 100\}$. The arrival rate of proactive DF request is $\lambda_P = \lambda_k = \lambda/K$. It can be seen that the exact and simulation results are very close, and under all conditions FF spectrum allocation method outperforms RF. For the regular system, the resource and fragmentation parts contribute to the overall blocking probability (Eq. (3.11)). For the blocking prob-

abilities of Pro-DF and Re-DF models (Eq. (3.10)), fragmentation and resource blocking are supplemented by defragmentation blocking, and for Pro-Re-DL-DF model (Eq. (3.9)), the fragmentation blocking is replaced by the defragmentation blocking. As expected, when average reconfiguration time ($T_{RT} \propto 1/\mu_d$) is reduced (i.e., rate is increased from 1 to 100), then the overall blocking probabilities also reduce for both FF and RF allocation methods. But the most important fact is that the defragmentation only results into an improved system performance, if the mean reconfiguration time is much smaller than the mean connection holding time. Only at lower reconfiguration time ($\mu_d = 100$), the overall blocking in all DF models is much lower than the regular system, and Pro-Re-DL-DF model outperforms other DF models for all μ_d. In contrast, if the mean reconfiguration time is in the range of the mean holding time, i.e. $\mu_d = 1$, the blocking performance of all DF models degrades, since higher reconfiguration time results in higher blocking (assumption (v)), and for medium to high load the defragmentation mechanism worsens the blocking probability if compared to the regular system. Another observation in this example is that the Re-DF model performs as good as Pro-DF model, irrespective of μ_d. The reason is that the arrival rate of proactive DF request is set to the arrival rate a class k request, i.e. $\lambda_P = \lambda_k$, and hence in Pro-DF model, DF is triggered from a fragmentation state after fragmentation detection time ($T_P \propto 1/\lambda_P$) which is nearly equal to the time interval when DF is triggered in Re-DF model by a class k connection request arrival in a fragmented state. However, we will see that the blocking performance of Pro-DF significantly improves with the increase in λ_P.

Next, we investigate the individual blocking types: RBP, FBP, and DBP for different models. In Table 3.3, we show the blocking parts under FF and RF policies due to the effects of lack of resources (Eq. 3.6), fragmentation (Eq. 3.7) and defragmentation

Table 3.3.: Resource, fragmentation, and defragmentation blocking (in %) for three different models under FF and RF spectrum allocation policies with EOL capacity C=20 slices, and demands $\boldsymbol{d} = (4, 6, 8)$ slices [37].

			Link-Load (Erlangs)																	
			2						6						10					
			RBP		FBP		DBP		RBP		FBP		DBP		RBP		FBP		DBP	
M*	μ_d	SA	Exact	Sim.	Exact	Sim.	Exact	Sim.	Exact	Sim.	Exact	Sim.	Exact	Sim	Exact	Sim	Exact	Sim	Exact	Sim
0	-	FF	0.86	0.86	0.28	0.28	0	0	6.53	6.52	2.21	2.21	0	0	14.10	14.10	4.45	4.45	0	0
		RF	0.64	0.64	2.31	2.32	0	0	4.49	4.49	9.15	9.15	0	0	9.62	9.63	14.78	14.79	0	0
I	1	FF	0.86	0.86	0.26	0.26	0.08	0.08	6.53	6.53	1.87	1.87	1.59	1.59	13.83	13.83	3.40	3.40	4.81	4.81
		RF	0.65	0.65	2.15	2.14	0.73	0.74	4.55	4.57	7.17	7.17	7.34	7.31	9.33	9.32	9.50	9.51	16.27	16.28
	100	FF	0.86	0.86	0.26	0.26	0.00	0.00	6.63	6.64	1.89	1.89	0.02	0.02	14.52	14.52	3.57	3.57	0.05	0.05
		RF	0.66	0.66	2.16	2.17	0.01	0.01	4.91	4.91	7.73	7.72	0.08	0.08	11.12	11.11	11.32	11.32	0.19	0.19
II	1	FF	0.86	0.86	0.26	0.26	0.09	0.09	6.54	6.54	1.79	1.78	1.82	1.79	13.84	13.83	3.18	3.19	5.44	5.47
		RF	0.66	0.66	2.09	2.09	0.97	0.97	4.60	4.58	6.49	6.50	9.50	9.48	9.36	9.36	8.15	8.15	19.81	19.81
	100	FF	0.86	0.86	0.26	0.26	0.00	0.00	6.66	6.65	1.82	1.82	0.02	0.02	14.63	14.62	3.36	3.37	0.06	0.06
		RF	0.66	0.66	2.11	2.11	0.01	0.01	5.08	5.06	7.17	7.18	0.10	0.11	11.64	11.65	10.15	10.14	0.25	0.24
III	1	FF	0.89	0.89	0	0	0.15	0.15	7.06	7.06	0	0	2.96	2.95	14.97	14.99	0	0	8.22	8.21
		RF	0.88	0.88	0	0	1.58	1.58	6.29	6.30	0	0	13.51	13.52	12.16	12.16	0	0	25.49	25.51
	100	FF	0.89	0.89	0	0	0.00	0.00	7.27	7.27	0	0	0.03	0.03	16.30	16.30	0	0	0.09	0.09
		RF	0.89	0.89	0	0	0.02	0.02	7.26	7.26	0	0	0.16	0.16	16.26	16.25	0	0	0.34	0.34

*Models M = 0, I, II ,and III for Regular, Pro-DF, Re-DF, and Pro-Re-DL-DF, respectively. μ_d is DF rate.

(Eq. 3.8). As expected, for the regular system ($M = 0$) and for traffic load range shown here, the blocking in RF policy is dominated by fragmentation and not by resource unavailability. In contrast, the FF policy shows the opposite behavior, where the resource blocking is quite larger than that of fragmentation blocking. This is due to the fact that the FF policy considerably reduces the number of fragmented spectrum patterns. Similarly, this explains the behavior already observed in Table 3.2, where all DF methods perform better for the RF policy, since the DF models reduce the fragmentation blocking, but as we also see, depending on the traffic load and reconfiguration rate μ_d this reduction may cause an increase of the resource blocking. This effect is shown for the RF policy by the shaded blocks, where we observe that with increasing traffic load the Re-DF model increasingly reduces the fragmentation blocking at the cost of worsening the resource blocking. This is more pronounced for the hybrid Pro-Re-DL-DF model, were the fragmentation blocking is completely omitted. More importantly, we also observe that resource blocking overtakes fragmentation blocking under RF policy in Re-DF model after some moderate load (here, at link-load of 10 Erl.). The additional blocking due to the defragmentation blocking has only an undesirably effect, if the defragmentation time is in the order of the connection holding time, e.g. for $\mu_d = 1$. Furthermore, at higher μ_d, which is the usual case in real networks ($T_{RT} \sim$ microseconds), this additional blocking is much lower than the other components and can be neglected.

Fig. 3.4 shows a lower bound on the defragmentation rate (μ_d) for a fixed load to get the positive blocking gain in Re-DF and Pro-Re-DL-DF models (Pro-DF performs like Re-DF for $\lambda_P = \lambda_k$) with respect to the regular system. This tells us about the minimum defragmentation rate (or maximum reconfiguration time) that a network operator can allow to get the lower blocking in our models than the regular system. This also gives an upper bound on the load up

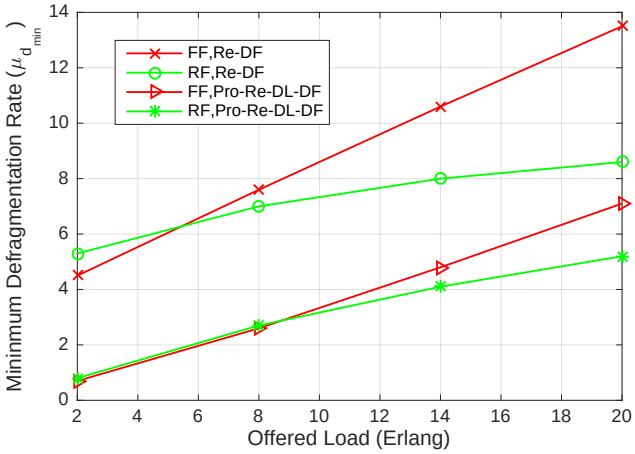

Figure 3.4.: Minimum defragmentation rate (μ_d) to get positive blocking gain under FF and RF spectrum allocation policies for Re-DF and Pro-Re-DL-DF models ($\lambda_P = \lambda_k, C = 20$ slices, $\boldsymbol{d} = (4, 6, 8)$ slices) with respect to the regular system [37].

to which the DF is effective (i.e., positive gain) when operated at a particular rate μ_d. One observation is that when load increases, defragmentation rate needs to be increased to maintain the positive gain due to defragmentation. Pro-Re-DL-DF model provides positive gain at relatively lower DF rate μ_d for all loads, and under RF policy, positive blocking gain can be achieved at relatively lower defragmentation rate at higher loads. In contrast, at lower loads, positive blocking gain can be achieved at lower or same μ_d under FF spectrum allocation policy.

Next, for the Pro-Re-DL-DF model, we show in Fig. 3.5 the average set-up delay time of requests in the control plane that could

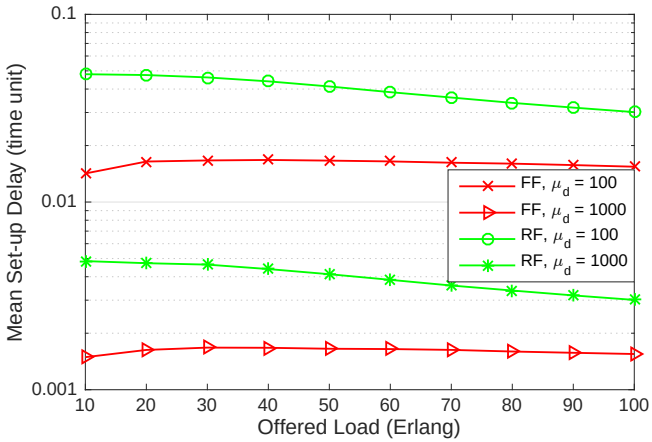

Figure 3.5.: The average set-up delay time in Pro-Re-DL-DF model under RF and FF spectrum allocation policy for C=100 slices, demands $\boldsymbol{d} = (4, 6, 8)$ slices, and $\lambda_P = 0$ [37].

not be set-up immediately due to the fragmentation of spectrum. In the Pro-Re-DL-DF model, such requests are delayed by the time until defragmentation is performed. As expected, the average waiting (delay) time is inversely proportional to the reconfiguration rate ($\tau_d \propto 1/\mu_d$), and it does not vary much by changing load. However, it should be noted that requests have to wait for slightly longer time under RF spectrum allocation policy. The reason is that under the RF spectrum allocation policy the mean defragmentation time is higher than the FF, since the connections are allocated randomly, and therefore the number of reconfiguration steps (η) is slightly higher.

3.4. Exact Network Model with Defragmentation

In this section, we relax two main assumptions of the DF models presented in the previous section 3.3, i.e., during reconfiguration period all new arrivals are blocked, and services of the existing connections are interrupted. Although two assumptions made DF models simpler, they resulted in limiting the blocking reduction. In this section, we assume that existing connections continue their services in the DF states, and if they complete their services (holding times) in a DF state, the system moves to one of least fragmented state(s) with an assumption that the DF has also been completed. The reason is that reconfiguration can be performed in much lower time ($\sim$ microseconds) with negligible or without any disruption of connections' services [46, 67], whose remaining mean service times in a DF state are $1/\mu_k$ ($\sim$ seconds or minutes). However, the effects of disruptions to existing traffic of reconfiguration can be severe even if the disruptions are microseconds long. If existing traffic is swept across during reconfiguration, the timing system of the receiver can be dislodged. Equalization of dispersion and polarization rotation must be readapted at the receiver. All these will have trigger link quality impairment. Even if it is not (as some papers has demonstrated), the loss of data in a few microseconds will be large enough to trigger TCP window closing if it is used as the transport layer protocol and other flow error control and recovery protocols will also trigger retransmissions. Nevertheless, in this thesis we do not consider the effect on higher layer protocol and also link impairment compensation upon reconfigurations.

Unlike the previous section, here we simply model the reconfiguration (DF) process with mean reconfiguration time $t_{DF} = 1/\mu_d$ which is independent from the number of reconfiguration steps required in DF states. As shown in Section 3.3, the reactive DF with a delayed admission outperforms other DF models, thus here we model a re-

active DF process (or simply DF) which allows a delayed connection setup after the reconfiguration period t_{DF}, and propose an exact CTMC model for evaluating the blocking in EONs with the DF process, which is *optimal* in a sense that after the reconfiguration a network experiences the least fragmentation possible, as defined by a fragmentation measure [65, 68]. However, unlike a single link, the measure of fragmentation in a mesh network by any metric can not be called optimal [69]. We introduce additional DF states for reconfiguration of connections in the exact CTMC model that we used for exact blocking analysis in EONs in the previous chapter with a fixed routing, i.e., $\kappa_o = 1$ for simplicity, and acknowledging the fact that the reconfiguration of a connection on alternate paths would be time consuming and it will disrupt the traffic flow. Therefore, the shortest route of an origin-destination (OD) pair $r(o, 1)$ is denoted by $r(o)$.

Model Assumptions

All notations and assumptions of the exact network model in Section 2.4 are also valid for the modeling the DF process in EONs under the fixed routing. The additional assumptions related to the DF process are given below.

- A DF process is triggered, when a request enters into a fragmented state where it could not be served normally.
- The request which triggers the DF process is put on hold (delayed), and it starts its service immediately after the defragmentation. The reconfiguration time is also exponentially distributed with mean $1/\mu_d$.

3.4.1. Generation of State-Space and State Transitions

The regular (i.e., without DF) network state-space Ω_V, related transitions and blocking states for a given classes of demands can be

obtained using the Algorithm 1 under a given spectrum allocation policy (e.g., RF, FF).

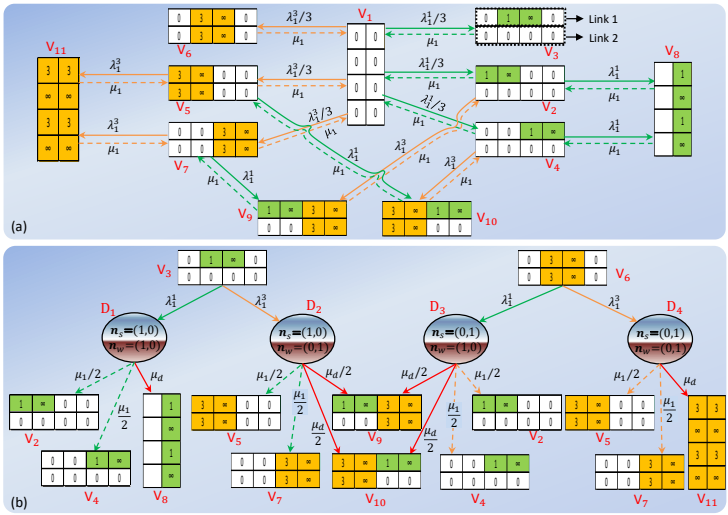

Figure 3.6.: (a) State transition diagram due to arrivals and departures of a demand $d_1 = 2$ slices under the RF policy in a 2-link network without DF with 4 slices per link and two OD pair routes $o = 1$ and $o = 3$. (b) All transitions leading to and from DF states in the same network with DF under the RF policy [68].

Fig. 3.6(a) shows all possible ($|\Omega_V| = 11$) regular states and related transitions due to arrivals and departures of 2-slice requests belonging to OD pair $o = \{1, 3\}$ ($r(o = 1) = 1 \rightarrow 2$, and $r(o = 3) = 1 \rightarrow 2 \rightarrow 3$) under the RF policy in a 2-link network (①→②→③) with $C = 4$ slices per fiber link. For simplicity, we assume that there is no arrival on a route $2 \rightarrow 3$ (i.e., link 2) in the 2-link network (i.e., $\lambda_1^2 = 0$). Observe that two network states V_3 and V_6 are fragmented

states for a 2-slice request of OD pairs $o = 1$ and $o = 3$, since these states have sufficient (2) free or empty slices on both links ($E(S^1) = 2$ and $E(S^3) = 2$) to accept a 2-slice demand on both routes, but they do not form a contiguous and continuous block. Thus, the states V_3 and V_6 would block 2-slice requests if spectrum is not defragmented.

Let us define a set of fragmented states for an arrival (o, k), i.e., for a class k arrival on an OD pair o, as

$$\mathbb{FI}(o,k) = \{V_i | f_l(\cup_j S^j) < d_k \leqslant \min(E(S^j)), j \in r(o), S^j \in V_i; \forall i\}. \tag{3.12}$$

The above set correctly identifies fragmented states for a class k OD pair request (o, k) by identifying network states that have sufficient number of free slices ($d_k \leqslant \min(E(S_j))$) on its route $r(o)$, but the maximum number of continuous and contiguous free slices (obtained by spectrum union operation in $f_l(\cup_j S^j)$) is not sufficient to accept a bandwidth demand d_k. Notice that when a route $r(o)$ does not have enough free slices (d_k) on any of its constituent links, i.e., $\min(E(S^j)) < d_k, j \in r(o)$ then the request (o, k) would be blocked immediately due to resource unavailability – referred as resource blocking – which is captured by a set $\mathbb{RB}(o, k)$ in step 5 of Algorithm 3. For $o = \{1, 3\}, k = 1, \mathbb{RB}(o, k) = \{V_8, V_9, V_{10}, V_{11}\}$, since these states are resource blocking states for class $k = 1$ requests with 2 slices demand on routes $r(o = 1)$ and $r(o = 3)$.

A spectrum fragmentation measure of a regular state V_i can be defined as

$$\phi_s(V_i) = 1 - \frac{\sum_{\forall(o,k)} |r(o)| \times |\Gamma_{V_i}^{o,k+}|^{RF}}{\sum_{\forall(o,k)} |r(o)| \times |\Gamma_{V_1}^{o,k+}|^{RF}} \tag{3.13}$$

where $|r(o)|$ is the number of hops on a route $r(o)$, and $|\Gamma_{V_i}^{o,k+}|^{RF}$ is the number of all possible ways (RF policy) by which a state V_i can allocate spectrum to a request (o, k). Notice that $|\Gamma_{V_1}^{o,k+}|^{RF} = C - d_k + 1$, and fragmentation measure of an empty state V_1 is 0.

We use (3.13) for measuring fragmentation of states under both RF and FF policies.

Let us now define a DF state D_t, wherein some of the connections are reconfigured, by the number of serving and waiting connections per class per route $\boldsymbol{n}_t^s = (n_{1,t}^{1,s},\ldots,n_{k,t}^{o,s},\ldots,n_{K,t}^{|\mathcal{O}|,s})$ and $\boldsymbol{n}_t^w = (n_{1,t}^{1,w},\ldots,n_{k,t}^{o,w},\ldots,n_{K,t}^{|\mathcal{O}|,w})$, respectively. DF states are basically created due to new arrivals which could not be accepted in fragmented regular states (steps 7–17), and possibly also due to arrivals in DF states (steps 18–32). For example in Fig. 3.6(b), a state V_3 can not accept a new arrival (o=1,k=1) immediately due to spectrum fragmentation, however, the DF-enabled network would delay the admission of this request, and perform reconfiguration of an existing 2-slice connection (o=1,k=1) in a DF state $D_1 \equiv (\boldsymbol{n}_1^s$=(1,0),$\boldsymbol{n}_1^w$=(1,0)). Notice that a regular network state V_i could be a fragmentation state for different classes of requests on different routes, leading to different DF states as shown in Fig. 3.6(b).

More importantly, it is not always possible to defragment or realign scattered free spectrum into a larger block of contiguous and continuous slices. For example, when a new single-slice OD pair connection o arrives on a route $3 \rightarrow 1 \rightarrow 2$ in Fig. 3.7(b), where two connections (with 1-slice bandwidth shown in green and violate colors) are allocated over slices s_1 and s_2 on their respective routes, respectively, then the connection request o is blocked even though there is a free slice on both links. In the current scenario, even an optimal DF process would not be able to reconfigure existing connections in order to accept the new request on its route, assuming that there is no spectrum converters. Therefore, unlike linear networks, blocking due to spectrum fragmentation can be reduced but not completely eliminated in a mesh network by a DF scheme. In case of a linear network in Fig. 3.7(a), a slice s_1 on both links can be allocated to a new 1-slice request on a route $1 \rightarrow 2 \rightarrow 3$, by shifting a connection from a slice s_2 to s_1 on a link 1. There-

Algorithm 3 Find Ω_V, Ω_D, transitions between states, and blocking states [68].

Given: RSA policy, Network topology, $C, \mathcal{J}, \boldsymbol{d}, \forall o : r(o)$.
Generate all exact network states Ω_V, a set of transition states $\Gamma^{o,k\pm}_{V_i}, \forall(o,k)$, and blocking states $\mathbb{B}(o,k)$ using the Algorithm 1.
for each V_i in Ω_V **do**
    Calculate fragmentation factor $\phi_s(V_i)$ as in Eq. (3.13).
    if $V_i \in \mathbb{B}(o,k)$ && $V_i \notin \mathbb{FI}(o,k)$ **then** $\mathbb{RB}(o,k) \leftarrow \{V_i\}$.
end for
// Create DF states and transitions due to arrivals in fragmented states
for $i = 1 : |\Omega_V|$ **do** // first initialize $\Omega_D = \emptyset, t = 1$
    for $(o,k) = (1,1) : (|\mathcal{O}|, K)$ **do** // $n^{o,w}_{k,t} = 1, n^{o,s}_{k,t} = 1$
        if $V_i \in \mathbb{FI}(o,k)$ && $(\boldsymbol{n}(V_i) + n^{o,w}_{k,t}) \in \boldsymbol{n}_{\Omega_V}$ **then**
            $D_t \leftarrow (\boldsymbol{n}^s_t = \boldsymbol{n}(V_i), \boldsymbol{n}^w_t = (..., 0, n^{o,w}_{k,t}, 0, ...))$
            if $D_t \notin \Omega_D$, then $\Omega_D \leftarrow [\Omega_D, D_t]$,
            $\Gamma^{o,k+}_{V_i} \leftarrow \{D_t\}, t \leftarrow |\Omega_D| + 1$.
        else if $V_i \in \mathbb{FI}(o,k)$ &&$(\boldsymbol{n}(V_i) + n^{o,w}_{k,t}) \notin \boldsymbol{n}_{\Omega_V}$
            then $\mathbb{FB}(o,k) \leftarrow \{V_i\}$
        end if
    end for
end for
//Initialize $t = 1$, Create DF states and related transitions due to arrivals in DF states
repeat
    $\mathbb{T}^+_{D_t} \leftarrow \{V_i | (\boldsymbol{n}^s_t + \boldsymbol{n}^w_t) = \boldsymbol{n}(V_i), \forall V_i \in \Omega_V\}$
    $\mathbb{T}^+_{D_t} \leftarrow \mathbb{T}^+_{D_t} \backslash \{V_i | \phi_s(V_i) > \min\{\phi_s(\mathbb{T}^+_{D_t})\}, \forall V_i\}$
    for $(o,k) = (1,1) : (|\mathcal{O}|, K)$ **do** // $n^{o,w}_{k,t} = 1, n^{o,s}_{k,t} = 1$
        if $(\boldsymbol{n}^s_t + \boldsymbol{n}^w_t + n^{o,w}_{k,t}) \in \boldsymbol{n}_{\Omega_V}$ **then**
            $\Gamma^{o,k+}_{D_t} \leftarrow D_\nu, D_\nu \leftarrow (\boldsymbol{n}^s_t, \boldsymbol{n}^w_t + n^{o,w}_{k,t})$.
            if $D_\nu \notin \Omega_D$ then $\Omega_D \leftarrow [\Omega_D, D_\nu], \nu \leftarrow |\Omega_D| + 1$.
        else
            $\mathbb{RB}(o,k) \leftarrow \{V_i\}$
        end if
        $\mathbb{T}^{o,k-}_{D_t} \leftarrow \{V_i | (\boldsymbol{n}^s_t - n^{o,s}_{k,t} + \boldsymbol{n}^w_t) = \boldsymbol{n}(V_i), \forall V_i\}$
        $\mathbb{T}^{o,k-}_{D_t} \leftarrow \mathbb{T}^{o,k-}_{D_t} \backslash \{V_i | \phi_s(V_i) > \min\{\phi_s(\mathbb{T}^{o,k-}_{D_t})\}, \forall V_i\}$
    end for
    $t \leftarrow t + 1$
until $t > |\Omega_D|$

fore, an arrival of a request (o, k) in a fragmented state V_i does not create a new DF state if there is no regular network states that can be reached after (re)allocating spectrum to existing connections $\boldsymbol{n}(V_i)$ and the new request (o, k), as shown in steps 13–14, where $\boldsymbol{n}_{\Omega_V} \equiv \{\boldsymbol{n}(V_1), \ldots, \boldsymbol{n}(V_{|\Omega_V|})\}$. Thus, such state V_i is identified as a fragmentation blocking state for requests belonging to (o, k) using a set $\mathbb{FB}(o, k)$.

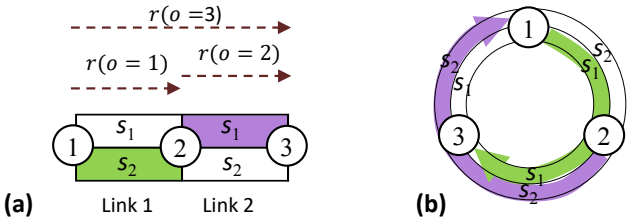

Figure 3.7.: Possibilities of defragmentation are shown in (a) 3 node (or 2-link) linear network, and in (b) 3 node ring network, with 2 slices per unidirectional fiber link.

A new DF state could also be created due to arrival of requests (steps 21–24) in a DF state D_t that has sufficient ($\geqslant d_k$) free slices on all traversing links on a requested route $r(o)$ to accommodate a new arrival (o, k), in addition to its waiting requests. After the completion of a reconfiguration process in a DF state D_t, the transition from D_t to one of possible least-fragmented regular network states occurs that have same set of connections ($\boldsymbol{n}(V_i) = \boldsymbol{n}_t^s + \boldsymbol{n}_t^w$), such that the objective of minimizing fragmentation is achieved (steps 19–20). For example in Fig. 3.6(b), after the completion of reconfiguration in a DF state $D_1 = (\boldsymbol{n}_1^s = (1, 0), \boldsymbol{n}_1^w = (1, 0))$, the system transits (shown by a red arrow) to only one possible least-fragmented state V_8 that has same set of connections $\boldsymbol{n}(V_8) = (2, 0)$. Furthermore, when a connection completes its service during the re-

configuration process in a DF state, it immediately accepts waiting requests by finishing the ongoing reconfiguration process and transits to a regular network state instead of going to (or creating) another DF state, see steps 28–29. This helps in limiting the total number of DF states $|\Omega_D|$. As an example, the departure of a connection in service ($o = 1, k = 1$) in the DF state D_1 leads to one of two least-fragmented regular states belonging to a set $\mathbb{T}^{1,1-}_{D_1} = \{V_2, V_4\}$ with rate $n^{1,s}_{1,1} \times \mu_1/|\mathbb{T}^{1,1-}_{D_1}| = \mu_1/2$, where the number of ($o = 1, k = 1$) serving connections in D_1 is $n^{1,s}_{1,1} = 1$.

The following transition function is defined to help map a DF state $D_t = (\boldsymbol{n}^s_t, \boldsymbol{n}^w_t)$ to a target network state V_i.

$$\beta(D_t, V_i) = \begin{cases} \mu_d/|\mathbb{T}^{+}_{D_t}| & \text{if } V_i \in \mathbb{T}^{+}_{D_t} \\ n^{o,s}_{k,t} \times \mu_k/|\mathbb{T}^{o,k-}_{D_t}| & \text{if } V_i \in \mathbb{T}^{o,k-}_{D_t} \\ 0 & \text{otherwise} \end{cases} \tag{3.14}$$

In (3.14), the function $\beta(D_t, V_i)$ selects the correct DF transition rate from a DF state D_t to a regular state V_i based on the completion of a DF process (with overall rate μ_d times the probability of transition into a least fragmented state $1/|\mathbb{T}^{+}_{D_t}|$) or due to departure of an (o, k) connection from the DF state D_t (with overall departure rate $n^{o,s}_{k,t} \times \mu_k$ times the probability of transition into a least fragmented state $1/|\mathbb{T}^{o,k-}_{D_t}|$). It should be noted that $\beta(D_t, V_i) = 0$ means that the state V_i is not a target state of the DF state D_t. Again, we use the LSQR method [61] to solve global balance equations and obtain steady state probabilities $\boldsymbol{\pi} = [\pi(V_1), \cdots, \pi(V_{|\Omega_V|}), \pi(D_1), \cdots, \pi(V_{|\Omega_D|})]$ for $V_i, i = 1, \ldots, |\Omega_V|$ and $D_t, t = 1, \ldots, |\Omega_D|$.

3.4.2. Blocking Analysis

The exact resource blocking probability (RBP) and fragmentation blocking probability (FBP) per class k per OD pair o in an EON with DF can be obtained by (3.15) and (3.16), respectively. The

RBP of (o,k) requests is calculated by summing over the regular (V_i) and DF (D_t) states which are resource blocking states for the (o,k) requests, i.e.,$\in \mathbb{RB}(o,k)$.

$$RBP_k^o = \sum_{i=1,V_i \in \mathbb{RB}(o,k)}^{|\Omega_V|} \pi(V_i) + \sum_{t=1,D_t \in \mathbb{RB}(o,k)}^{|\Omega_D|} \pi(D_t) \tag{3.15}$$

The FBP of (o,k) requests happens due to the fragmentation, thus it can be calculated by summing over the states which are fragmentation states $\mathbb{FB}(o,k)$ for the (o,k) requests.

$$FBP_k^o = \sum_{i=1,V_i \in \mathbb{FB}(o,k)}^{|\Omega_V|} \pi(V_i) \tag{3.16}$$

The overall blocking probability in an EON with DF is given by ensemble averaging over sum of resource and fragmentation blocking probability $(RBP_k^o + FBP_k^o)$ of all classes $k, k = 1, \ldots, K$ on all OD pair requests $o \in \mathcal{O}$.

In EONs without DF, (3.15) and (3.16) can be used to obtain RBP and FBP provided state probabilities are obtained without DF states, $\Omega_D = \emptyset$. Thus, only steps 1 to 6 need to be executed in Algorithm 3. We evaluate the Exact DF network model along with the approximate DF models, which is presented next, in Section 3.5.3.

3.5. Reduced State Occupancy DF Models

In this section, we present a reduced EON model with DF in order to tackle the scalability issue of the exact DF blocking model. To this end, we utilize the reduced state model presented in the previous chapter and present approximate blocking analysis for a link and EONs with DF.

3.5.1. Reduced DF Link Model

To reduce the number of regular and DF states in an elastic optical link (EOL), let us represent a regular link state (say microstate) by a random variable representing total link occupancy $X = x$, and a DF state by $X_d = (x_s, x_w)$, where x_s and x_w are the serving bandwidth (in slices) and additional bandwidth required by the waiting connection requests, respectively. In the reduced DF link model, we can accept class k requests that arrive in non-blocking exact link states $\mathbb{NB}(x,k)$, and fragmented exact link states $\mathbb{FI}(x,k)$, having total occupied slices x, as defined by Eqs. (2.6) and (2.7), respectively.

Unlike the mesh network, a link ensures that a demand which could not be accepted due to fragmentation of spectrum gets accepted after the DF operation. Therefore, in our reduced DF link model, the probability that a class k request will be accepted due to defragmentation in a microstate x is given by $g_k^{App.}(x;\bar{x}) = 1 - p_k^{App.}(x;\bar{x}), 0 \leqslant x \leqslant C - d_k$ and 0, otherwise. Using this relation, we can obtain $g_k^{App.}(x;\bar{x})$ for both *App.EES* (using Eq. (2.14)) and *App.SOC* (using Eq. (2.15)) as follows.

$$g_k^{App.EES}(x) = \frac{|\mathbb{FI}(x,k)|}{|\Omega_S(x)|} \tag{3.17}$$

$$g_k^{App.SOC}(x;\bar{x}) = \frac{|\mathbb{FI}(x,k)|}{|\Omega_S(x)|} \times \left(1 - exp\left(-\frac{\bar{x}}{C} \times |log(\frac{x}{\bar{x}})|\right)\right) \tag{3.18}$$

Note that above equations use this relationship: $|\Omega_S(x)| = |\mathbb{NB}(x,k)| + |\mathbb{FI}(x,k)|$, for $0 \leqslant x \leqslant C - d_k$. The above equations are true for resource blocking microstates as well, since $|\mathbb{FI}(x,k)| = 0$ for $C - d_k < x \leq C$. Furthermore, the overall probability of acceptance is $p_k^{App.}(x;\bar{x}) + g_k^{App.}(x;\bar{x}) = 1$, for $0 \leqslant x \leqslant C - d_k$, and 0 otherwise. Therefore, for both approximations the connection setup rate in a microstate x is $\alpha_k(x) = \lambda_k, 0 \leqslant x \leqslant C - d_k$, and 0 otherwise. Fig. 3.8 shows the transitions among regular microstates and DF states for the RF policy using *App.EES*. The arrival of a class k request in

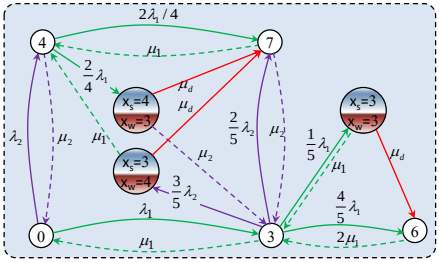

Figure 3.8.: A state transition diagram with DF microstates in a 7-slice fiber link with two classes of demands $\boldsymbol{d} = (3, 4)$ slices under the RF policy.

a microstate x triggers the system to transit to a DF state (x_s, x_w) with rate $\lambda_k \times g_k^{App.EES}(x) = \lambda_k \times \frac{|\mathbb{FI}(x,k)|}{|\Omega_S(x)|}$. A new arrival, departure, or completion of a DF process would transit the system to the respective microstates with rates λ_k, μ_k and μ_d, respectively. For example, in Fig. 3.8 a class 2 arrival (4-slice) in a microstate $x = 3$ would lead the system to a regular occupancy state with probability $2/5$ and to a DF state $(x_s = 3, x_w = 4)$ with probability $3/5$, due to the fact that among the five possible link-states with occupancy $x = 3$ there are 2 non-blocking and 3 fragmented states for class 2 demands (see Table 2.3). Notably, the DF process eliminates fragmentation blocking in a linear network (e.g., a link), and blocking is only due to resource unavailability. However, this gain is obtained at the cost of delaying the admission of some of the connections.

Let us define the following transition function to help map a DF microstate $X_d = \mathbf{x}_d$ to a target microstate $X = x$.

$$\beta(\mathbf{x}_d, x) = \begin{cases} \mu_d & \text{if } x = x_s + x_w \\ E[n_k|x_s] \times \mu_k & \text{if } x = x_s + x_w - d_k \\ 0 & \text{otherwise} \end{cases} \tag{3.19}$$

In Eq. (3.19), the function $\beta(\mathbf{x}_d, x)$ selects the correct DF transition

rate from a DF state to a regular microstate based on the completion of a DF process (with rate μ_d) or a departure (with rate μ_k) resulting in no serving connections. It should be noted that $\beta(\mathbf{x}_d, x) = 0$ means that the microstate x is not a target state of the DF microstate $\mathbf{x}_d$. Using the above transition identifying functions, we can solve the GBEs of all regular (x) and DF ($\mathbf{x}_d$) microstates in the reduced link model to obtain the steady state link occupancy distribution $\pi(x)$ and $\pi(\mathbf{x}_d)$.

The class k blocking probability in a DF-enabled EOL is

$$BP_k = \sum_{x=C-d_k+1}^{C} \pi(x) + \sum_{(x_s,x_w)|x_s+x_w+d_k>C} \pi(x_s, x_w). \tag{3.20}$$

In (3.20) blocking happens in such microstates (including DF) which do not have enough free slices, thus the resource blocking probability $RBP_k = BP_k$.

It should be noted that the waiting list of connection requests vary per DF state, since new arrivals are delayed during DF processes. Moreover, the addition of DF states to the regular microstates increases the total number of states and the complexity of the reduced DF link model. Thus, for intractability reason the following subsection proposes a reduced state DF network model with an assumption that the mean reconfiguration time is zero (this is inline with practical network reconfiguration time, $\sim$ microseconds), i.e., requests could be immediately accepted into fragmented states.

3.5.2. Reduced DF Network Model

The reduced state DF network model utilizes the same reduced state network model presented in Section 2.6, since we assume that the reconfiguration time is zero and therefore there are no DF microstates. Additionally, all notations, parameters, assumptions and both RSA constraints of the exact blocking model with DF are valid. Here, we

utilize the fixed routing approach. The only difference with the reduced network model (i.e., without DF) is that under a DF-enabled network, the probability of acceptance of a request in a given route occupancy is computed differently.

In a DF scenario, the overall probability of acceptance term ($p^{App.}_{k,DF}(\mathbf{x}_r;\bar{\mathbf{x}}_r)$) is more challenging to obtain accurately. The reason is that even though sufficient free, but scattered, spectrum is available on a route in a mesh network, connections can not be reconfigured always to accept a new request on the route. Therefore, only a fraction of arrivals into fragmented states could be accepted in EONs with DF. The DF network model needs to take into account the probability of acceptance term without DF (given by first product term), and the probability of acceptance due to a DF process (second product term) in only such constituting links ($j_i, i = 1,\ldots,h$) of a route r that are in fragmentation states, which is given by the sum of two product-form approximations as

$$p^{App.}_{k,DF}(\mathbf{x}_r;\bar{\mathbf{x}}_r) = \prod_{j_i\in r} p^{App.}_k(x_{j_i};\bar{x}_{j_i}) + \prod_{j_i\in r:|FB(x_{j_i},k)|\neq 0} g^{App.}_k(x_{j_i};\bar{x}_{j_i}). \tag{3.21}$$

Note that in Eq. (3.21), the overall probability of acceptance $0 \leqslant p^{App.}_{k,DF}(\mathbf{x}_r;\bar{\mathbf{x}}_r) \leqslant 1$, since the RHS of Eq. (3.21) is less or equal to $\prod_{j_i\in r}(p^{App.}_k(x_{j_i}) + g^{App.}_k(x_{j_i})) \leqslant 1$.

The RBP and FBP per OD pair $o \in \mathcal{O}$ per class k in an EON with or without DF is obtained by (3.22) and (3.23), respectively. Notice that RBP is computed over all route occupancy states where a class k request does not find enough free spectrum d_k, i.e., $C - d_k + 1 \leqslant x_{j_i} \leqslant C$ on at least one link of the route, $j_i \in r(o)$). Equivalently, it is calculated by one minus probabilities of observing the route with link state occupancies lying in $0 \leqslant x_{j_i} \leqslant C - d_k$. The FBP is simply compute by total blocking, given in Eq. (2.31), minus the RBP,

resulting in Eq. (3.23).

$$RBP_k^o = 1 - \sum_{x_{j_1}=0}^{C-d_k} \cdots \sum_{x_{j_l}=0}^{C-d_k} \pi_{j_1}(x_{j_1}) \cdots \pi_{j_l}(x_{j_l}) \tag{3.22}$$

$$FBP_k^o = \sum_{x_{j_1}=0}^{C-d_k} \cdots \sum_{x_{j_l}=0}^{C-d_k} \pi_{j_1}(x_{j_1}) \cdots \pi_{j_l}(x_{j_l}) \times [1 - p_k(\mathbf{x}_r; \bar{\mathbf{x}}_r)] \tag{3.23}$$

The overall blocking probability per class per OD pair is $RBP_k^o + FBP_k^o$. In EONs with (or without) DF, $p_k(\mathbf{x}_r; \bar{\mathbf{x}}_r)$ is replaced by $p_{k,DF}^{App.}(\mathbf{x}_r; \bar{\mathbf{x}}_r)$ (or $p_k^{App.}(\mathbf{x}_r; \bar{\mathbf{x}}_r)$). The blocking parts in (3.22) and (3.23) use steady state probability distribution $\pi_j(x_j)$, which is obtained by solving GBEs in (2.30) for all regular valid microstates x_j in $0 \leqslant x \leqslant C$ on each link $j \in \mathcal{J}$, in addition to $\sum_{x=0}^{C} \pi_j(x_j) = 1$, where again connection setup rates in Eq. (2.26) use $p_{k,DF}^{App.}(\mathbf{x}_r)$ (or $p_k^{App.}(\mathbf{x}_r)$) for computing blocking in EONs with DF (without DF) using the fixed-point iteration methodology described in Section 2.6.

3.5.3. Numerical and Simulation Results

In this section, we first analyse individual blocking parts (RBP and FBP) obtained by the exact EON model with and without (w/o) DF under two different spectrum allocation policies, RF and FF, and compare them with the discrete event simulation results for a unidirectional 2-link network and a unidirectional 3-node ring network. Furthermore, we show the accuracy of the reduced state model by comparing approximate RBP and FBP with the exact results in a small scale link ($C = 20$ slices), and with the simulation results in a medium and large scale 14-node NSFNET topology. The App.EES and APP.SOC approximations are used to obtain approximate blocking probabilities under the RF and FF policies, respectively. Blocking results are depicted versus *offered load*, which is defined as $\sum_o \sum_k \frac{\lambda_k^o}{\mu_k}$, where $o \in \mathcal{O}, k = 1, 2, \cdots, K$. We assume that

the service (holding) times of connection requests between an OD pair are exponentially distributed with mean $1/\mu_k = 1$ unit, and per-class, per OD pair connection requests arrive according to a Poisson process with uniformly distributed rate λ_k^o= *offered load*$/(|\mathcal{O}| \times K)$. The reconfiguration time (t_{DF}) is exponentially distributed with a small mean $1/\mu_d = 10^{-3}$ time unit, unless otherwise stated. We present the average BP considering all OD pairs o and classes k. All results presented here consider both the spectrum contiguity and continuity constraints. We generated 10^7 connection requests to simulate EONs with or w/o DF. It should be noted that the simulation of the DF operation is not easy in multi-hop EONs, since the RSA constraints must be satisfied during the connection reconfiguration process, and reconfiguration may not be optimal. Furthermore, when a route is defragmented to create a new lighpath for admitting a connection request, all lightpaths ideally might have to be considered in conjunction with how cross traffic in the individual links are affecting other lightpaths that shares some but not all links with the new lightpath. This tends to couple the scheduling of all the links in a mesh network which will have very high computational and time complexity, and connections' data need to be buffered during DF, which is basically impracticable if more connections have to be buffered at Gigabit rate. Thus, for the simulation, we adopt a graph approach [60], explained in 4, to construct an auxiliary graph to find a possible solution in non-polynomial time. The auxiliary graph transforms the DF problem into a matter of finding non-overlapping set of free spectrum paths for the connections to be reconfigured in order to create a free spectrum block for an incoming demand which could not be accepted w/o DF. Thus, we simulate a reactive DF method, which is performed in zero time in the event simulation, and the request which cause the DF is accepted immediately if the DF function finds a sufficient free block of free slices.

Figs. 3.9 and 3.10 present the exact blocking parts (FBP and

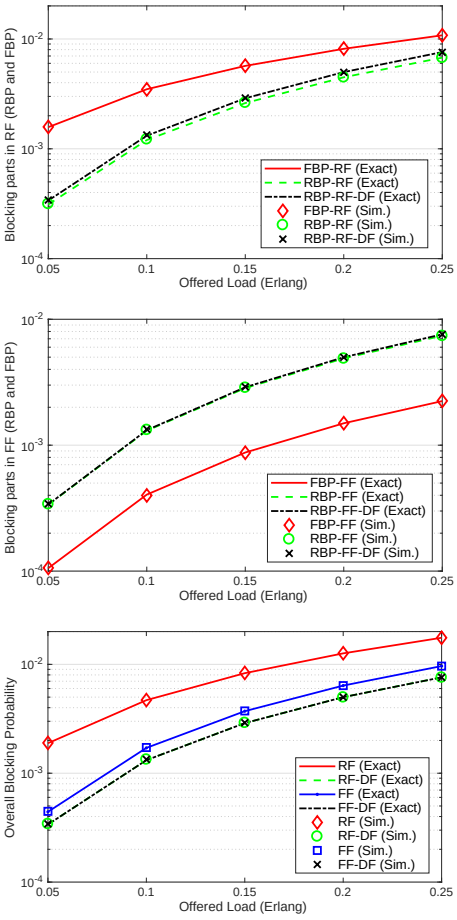

Figure 3.9.: Exact and simulation blocking results with and w/o DF in a 2-hop network with 3 OD pair routes with $C = 10$ and demands $d_k = \{3, 4\}$ slices. (top) Individual BP parts under the RF policy; (middle) Individual BP parts under the FF policy; and (bottom) Overall BP under both policies.

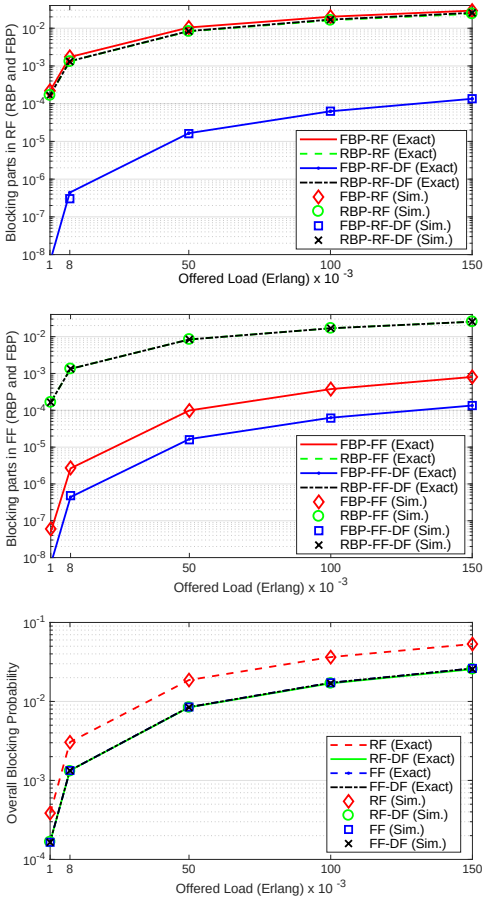

Figure 3.10.: Exact and simulation blocking results in a 3-node ring network with 6 OD pair routes with $C = 7$ and demands $d_k = \{3, 4\}$ slices. (top) Individual BP parts under the RF policy; (middle) Individual BP parts under the FF policy; and (bottom) Overall BP under both policies.

RBP) and also overall blocking probability in a 2-link network with 3 OD pair routes and in a 3-node ring network with 6 OD pair routes, respectively, under the RF and FF policies. First, we observe that individual blocking parts and overall blocking obtained by the exact Markov models with and w/o DF are very accurate, as they match with the simulation (Sim.) results. Second, under the RF policy blocking due to fragmentation (FBP-RF) is a major contributor to the overall BP, and it is higher than the blocking due to resource unavailability (RBP-RF) for the shown loads where blocking remains below 10% in a regular EON w/o DF. Third, the FBP with the DF operation can be either eliminated (see Fig. 3.9) in a linear EON or reduced as compared to FBP w/o DF in mesh networks (see Fig. 3.10) under both RF and FF policies. Although the overall blocking can be reduced to the same level under both policies by utilizing the DF strategy, the RF is more beneficial, since the FBP is much higher than the RBP in the RF policy in w/o DF operation (Fig. 3.9). On the other hand, the RBP is dominating the FBP in the FF policy with or w/o DF scenario in these small scale networks, where blocking reduction due to DF is negligible due to lower link capacity. It is important to mention that in Fig. 3.9 the simulation under the DF scenario does not completely eliminate fragmentation blocking the 2-link example, however, it is not shown, since the values are very small ($< 10^{-5}$). The reason is that the exact DF model is a so-called *optimal*, but not the simulation. Another important point is that even under a hypothetical optimal DF model, which employs an optimal fragmentation metric to define least-fragmentation states, there will be non-zero FBP in mesh networks (e.g., ring) under the RF and FF policies, since the fragmentation blocking (or deadlock) states will be visited with non-zero state probabilities under both of these policies which come under a greedy approach. A greedy spectrum allocation method accepts all requests if free resources can be assigned. On the other hand, a non-greedy approach rejects some

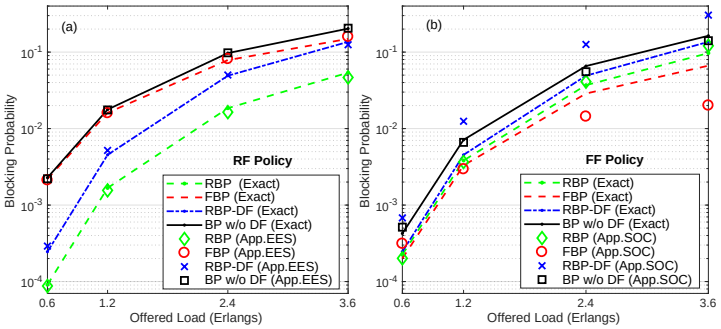

Figure 3.11.: Exact and App. RBP and FBP in an EOL with $C = 20, \boldsymbol{d} = (3, 4)$.

of the requests even though free resources are sufficient to allocate. Thus, we suggest that one should consider a non-greedy spectrum allocation policy in order to avoid visiting the deadlock states that can not be defragmented. This will possibly eliminate fragmentation blocking under the DF in all kinds of network topologies.

The accuracy of the reduced DF link model presented in Section 3.5.1 is evaluated in Fig. 3.11 in terms of approximate (App.EES for RF and App.SOC for FF) RBP and FBP in an EOL, and they are compared to the exact blocking results in the link with and w/o DF under both the RF and the FF policies. In Fig. 3.11 we observe that the App.EES blocking results match closely to the exact results under the RF policy for both with and w/o DF operations. However, the approximate FBP obtained by the App.SOC in EONs without DF are underestimated at higher loads, and the approximate RBP (which is also the overall BP) under the DF scenario is slightly overestimated. The reason is that the transition rate from a regular microstate to another microstate with probability $p_k^{App.SOC}(\cdot)$ and/or to a DF microstate with probability $g_k^{App.SOC}$ might not be

very accurate in the App.SOC approach. Nevertheless, in the DF-enabled link, the overall exact (or App.) blocking is given by the RBP (where the FBP is zero), and they are nearly same under both policies, as class k requests are accepted into the fragmented states $\mathbb{FI}(x, k)$.

Next, we present approximate RBP and FBP in the NSFNET topology with and w/o DF scenarios under the RF and FF polices for two different capacities and demands in Table 3.4, and verify them with the simulation results obtained under both the RF and FF policies. As explained in the Section 3.5.2, the mean reconfiguration time t_{DF} is assumed to be zero. Unlike the exact blocking results obtained in small networks, for both network scenarios $C = 100, d_k = \{3, 4, 6\}$ and $C = 200, d_k = \{4, 6, 10\}$ in NSFNET, we observe that the FBP is a major contributor in the overall blocking under both the RF and the FF policies w/o DF, which can be reduced significantly up to moderate loads by applying a reactive DF scheme with delayed acceptance of requests. Furthermore, Approximate (App.EES) RBP and FBP are very close to the corresponding simulation results underestimated in the RF w/o DF case. However, notice that the RBPs obtained by simulation under the RF policy in $C = 200$ case for all shown loads are zero (because the number of simulated events 10^7 might not be enough), and the corresponding approximate RBPs are also very small. On the other hand, in the RF with DF (i.e., RF-DF) case, the App. RBP is underestimated, but the App. FBP is overestimated. Interestingly, the blocking parts obtained by the App.SOC under the FF policy with or without DF is close to simulation results. Acknowledging the fact that in a large network with DF the simulation only provides suboptimal solutions, the overall blocking obtained by the reduced state model is close enough to the simulation results, thus the App.EES and App.SOC approximations are acceptable. More importantly, by applying DF the FBP is reduced at the cost of increasing RBP,

Table 3.4.: Resource and fragmentation BP parts in a 14-node NSFNET under the RF and FF policies with and without DF [68].

Scenarios	$C = 100, d_k = \{3, 4, 6\}, t_{DF} = 0$ time unit											
	offered load = 150				offered load = 200				offered load = 250			
	RBP		FBP		RBP		FBP		RBP		FBP	
	Sim.	App.	Sim.	App.	Sim.	App.	Sim	App.	Sim	App.	Sim	App.
RF	5.0e-7	8.0e-9	3.6e-2	2.5e-2	9.1e-7	1.0e-7	8.9e-2	7.0e-2	3.0e-6	5.4e-7	1.5e-1	1.2e-1
FF	2.7e-5	2.5e-4	1.6e-2	1.6e-2	6.1e-5	2.5e-3	6.3e-2	5.0e-2	8.8e-5	1.1e-2	1.2e-1	9.5e-2
RF-DF	8.5e-4	4.0e-4	3.7e-3	1.2e-2	5.3e-3	1.3e-3	2.1e-2	4.8e-2	1.3e-2	3.1e-3	5.4e-2	9.4e-2
FF-DF	1.0e-3	1.9e-3	3.4e-3	2.8e-3	6.2e-3	1.3e-2	1.9e-2	1.2e-2	1.5e-2	3.9e-2	5.2e-2	2.3e-2
	$C = 200, d_k = \{4, 6, 10\}, t_{DF} = 0$ time unit											
RF	0.0	1.7e-13	9.2e-3	1.2e-3	0.0	1.3e-12	3.6e-2	1.1e-2	0.0	1.1e-11	7.4e-2	3.4e-2
FF	1.7e-6	2.4e-6	7.9e-4	8.5e-4	7.9e-6	7.8e-5	1.1e-2	8.4e-3	1.9e-5	6.7e-4	3.8e-2	2.9e-2
RF-DF	1.1e-5	2.3e-7	1.8e-4	8.5e-4	2.9e-4	2.0e-6	2.8e-3	8.7e-3	1.8e-3	1.0e-5	1.3e-2	3.0e-2
FF-DF	4.0e-5	2.2e-5	8.7e-5	2.6e-4	7.6e-4	6.4e-4	1.8e-3	3.1e-3	3.9e-3	4.7e-3	9.4e-3	1.1e-2

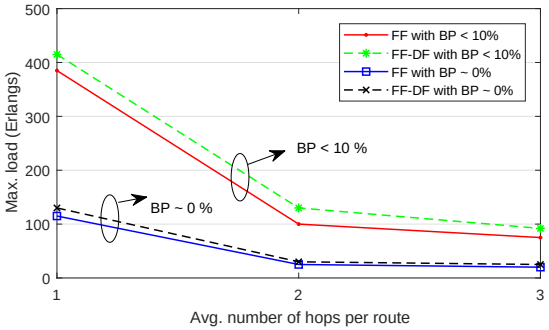

Figure 3.12.: Maximum carried load for various BPs under FF and FF-DF in a fully connected 5-node network with $C = 100, \boldsymbol{d} = (3, 4, 6)$.

however, the overall blocking remains lower under the DF operation, which contradicts the observation in Section 3.3.2, where we showed that the overall blocking could increase under the DF operation with a higher reconfiguration time. This was because of the model assumptions where DF states were assumed to block all arriving requests and interrupt services of existing connections, which indirectly increased holding times of connections.

Finally, Fig. 3.12 depicts the effect of increasing number of hops per route on the maximum (max.) load carried under the FF policy with and without DF in a fully connected 5-node network. The max. carried load is obtained through the simulation for BP less than 0.1 (10%) and for BP less than 10^{-6} ($\approx 0\%$). We observe that the network carries maximum load when average (avg.) number of hops per route is minimum (i.e., 1), since the spectrum continuity constraint is not required. Furthermore, DF is more beneficial in networks with routes having smaller number of hops, as overlapping lightpaths (or cross traffic) are lower.

3.6. Summary

In this chapter, we first modeled the occupancy of an elastic optical link under the various defragmentation approaches. We considered a proactive, a reactive and a combined DF methods with the acceptance of delayed requests. Then, we proposed the first exact Markov model for analyzing blocking probability in EONs under the reactive defragmentation with the acceptance of delayed requests, and subsequently a method to reducing the exact model into a reduced occupancy model to computing approximate blocking in EONs with DF. With both exact and approximate models we showed analytically that spectrum fragmentation is a major cause of connection blocking in EONs upto moderate loads under both popular spectrum allocation policies, random-fit and first-fit. Thus, we applied defragmentation as a connection reconfiguration method, and model it analytically to show its benefit in terms of reducing blocking under both policies. The numerical results obtained show that the exact analysis of resource and fragmentation blocking is accurate, albeit limited to a very small scale EONs, due to complexity. Finally, the approximate solutions have been shown useful in a broader range of scenarios, and the accuracy depends on the various factors, e.g., spectrum allocation policy, link capacity, and traffic loads. The next steps in this line of work include major challenges that have not been solved yet, and are analytically rather complex, most notably the interdependency of network link loads, and recognizing, differentiating and reserving resources separately for short, medium and long lightpaths in order to reducing the need of defragmentation.

4

Efficient Resource (Re)allocation Schemes in Elastic Optical Networks

4.1. Introduction

The analytical work presented in the previous two chapters is useful for network dimensioning under the RF and FF spectrum allocation schemes, but they are simple and neither allocate resources according to the current state of network fragmentation nor exploit application-dependent features. To allocate the optical resources efficiently in order to increase spectrum utilization or reduce blocking probability in EONs, there has been significant amount of work that exploited various optical resources, such as bandwidth variable transponders and cross-connects, fibers, spectrum, modulation formats, and traffic properties, such as bandwidth, holding time, start and end of connections, if known or through estimation methods when unknown. Therefore, researchers proposed different routing, spectrum, modulation, and core allocation schemes for efficient resource allocation in EONs, as surveyed in [30].

Whether it is single- or multipath routing, the flexible assignment of spectrum in a dynamic connection setup and tearing-down environment leaves the gaps between occupied spectrum, which might not be used by higher bandwidth demands due to unavailable con-

tiguous and continuous free spectrum. Thus, the fragmentation of free or occupied spectrum is a major problem in EONs that leads to reducing spectrum utilization and increasing connection blocking probability. To this end, numerous fragmentation-aware routing and spectrum allocation (RSA) have been proposed, and a detailed survey is provided in [31].

As illustrated in the previous chapters, the non-uniform bandwidth demands leads to spectrum fragmentation in a dynamic environment. Thus, arrival and departure times (or simply holding times) of connections could play a vital role in managing spectrum resources. Holding-time-aware-based RSA schemes are shown to be useful in minimizing fragmentation and connection blocking [28, 29]. However, the reconfiguration of existing connections by means of a spectrum reallocation strategy, which is often called as spectrum defragmentation, is absolutely necessary to keep spectrum fragmentation under control in a dynamic environment, even if we employ a fragmentation-aware or holding-time-aware spectrum allocation schemes [31].

This chapter presents heuristic solutions to address efficient resource allocation and reallocation schemes in elastic optical networks. We handle the fragmentation not only in the spectral and spatial domains but also in time domain, since connections usually have different holding (service) times, and they traverse on different paths in the network. We first formulate a network fragmentation metric considering spectral, spatial and time domains. Furthermore, we propose a holding-time-aware resource allocation scheme to assign connections to least fragmented spectrum paths. Finally, we propose a proactive as well as a reactive spectrum defragmentation schemes to efficiently reallocate spectrum resources to the existing and incoming connection demands.

4.2. Supporting Publications

- S. K. Singh and A. Jukan, "Efficient Spectrum Defragmentation with Holding Time Awareness in Elastic Optical Networks," *IEEE/OSA Journal of Optical Communications and Networking,* vol. 9, issue 3, pp. B78-B89, 2017. `https://doi.org/10.1364/JOCN.9.000B78`.

- S. K. Singh and A. Jukan, "Holding-Time Information (HTI): When to Use it?," in *Optical Fiber Communication (OFC) Conference,* Los Angeles, California, USA, March 2017.

- S. K. Singh and A. Jukan, "Non-Disruptive Spectrum Defragmentation with Holding-Time Awareness in Optical Networks," in *Optical Network Design and Modeling (ONDM),* Spain, May 2016.

4.3. Fragmentation Metrics

Most of the past work mainly considered spectral and spatial dimensions to measure spectrum fragmentation. However, as mentioned before, spectrum fragmentation in time domain also needs to be considered. Recently, [28,29] have applied the knowledge of holding-times or remaining lifetimes of connections to measure fragmentation. When the holding times of connections are known, it is easy to find remaining lifetimes of connections. On the other hand, when holding times are unknown, the estimated value of mean remaining lifetime of connections could to be used, as proposed for WDM networks using exponentially distributed holding time in [70]. In order to correctly measure the spectrum fragmentation in a fiber-link, we focus on the definition of *fragmentation metrics* in two domains, spectral and time. To this end, we first extend the existing approaches for measuring fragmentation in spectral domain. After that, we propose a new fragmentation metric in time domain, and a

combined metric is proposed for a network considering an additional spatial dimension in the form of network links that can be used to determine the time-point for triggering reactive defragmentation.

4.3.1. Fragmentation Metric – Spectral and Spatial Domains

Let us first define a spectral fragmentation metric (or FI), similar to [71] and [36], as the ratio between the sum of the set of bandwidth requests that can be accepted in a fragmented link and the set of bandwidth requests that can be accepted in a non-fragmented link. In addition to that, let us take into account all possible ways that a bandwidth request b_i with demand of i slices can be allocated in a free spectrum block of size n, i.e. $n-i+1$ number of possible ways in which spectrum can be allocated to this request b_i. Furthermore, let us consider all demands that can be accepted into the system. A spectrum block of n free slices can accommodate requests with bandwidth $\{b_1, b_2, \ldots, b_n\}$, where a bandwidth request b_i requires i number of consecutive free slices. A free block of n slices can accommodate a request b_i in $n-i+1$, where $1 \leq i \leq n$, possible ways in i consecutive slices. Therefore, total bandwidth assigned in all possible ways is $\sum_{i=1}^{n_k}(n_k - i + 1) \times b_i$, where n_k is the number of consecutive free slices in a k^{th} free block. Note that for $b_i = i$ and $i = 1$ to n_k, this summation can be simplified to $\binom{n_k+2}{3}$. If a link contains K such free blocks then total bandwidth that can be assigned is $\sum_{k=1}^{K}\sum_{i=1}^{n_k}(n_k - i + 1) \times b_i$. However, if these blocks were not separated apart, i.e., if all free slices are contiguous, then the total assigned bandwidth (in all possible ways) in $\sum_{k=1}^{K} n_k = N$ (say) free slices is equal to $\sum_{i=1}^{N}(N - i + 1) \times b_i$. Finally, we define the FI of a link l in the spectral domain as follows.

$$\phi_s(l) = 1 - \frac{\sum_{k=1}^{K}\sum_{i=1}^{n_k}(n_k - i + 1) \times b_i}{\sum_{i=1}^{N}(N - i + 1) \times b_i} \tag{4.1}$$

Consider a simple example of a fiber-link occupancy in Fig. 4.1(a)

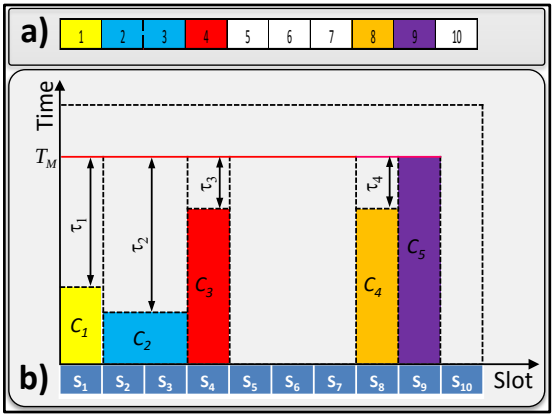

Figure 4.1.: Spectrum fragmentation in a 10-slice fiber link: (a) spectral domain and (b) time domain [60].

to illustrate this new metric. Here, spectral FI of this link can be calculated as follows. The link contains two blocks with 3 and 1 consecutive free slices. Therefore, $\phi_s(l)$ is 0.45 $\left(= 1 - \left(\binom{5}{3} + \binom{3}{3}\right) / \binom{6}{3}\right)$. The computing complexity of the above metric depends on the number of fragments on a given link l and the classes of bandwidth demands. As we consider all possible demands and each connection requires at least one spectrum slice, the maximum number of fragments is C/2, where C is the number of spectrum slices, i.e., capacity of a fiber link. Therefore, the time complexity for calculating fragmentation metric in spectral domain is O(C). The spectral fragmentation metric in Eq. (4.1) can be extended to consider a spatial dimension, by simply adding spectral fragmentation in multiple links on a route as well as on all fiber-cores per link in space-division multiplexing-based EONs, as we show in [72]. In this chapter, we evaluate EONs with single-core fiber links. We consider the spatial-division multiplexing with multi-core fibers in Chapter 5.

4.3.2. Fragmentation Metric – Time Domain

Since we consider dynamic network conditions, spectrum fragmentation in time domain affects only future arrivals, not the current arrival. This is because when a connection request R arrives at time t_i, spectrum occupancy in spectral (and not the time) domain at the time t_i determines if that request can be accepted into the system immediately or not. However, connections leave the system (after their remaining lifetimes $\bar{r}_i$) at different instances of time $t'_i = t_i + \bar{r}_i, \forall i$, thus, possibly fragment the spectrum in spectral domain. Therefore, time domain spectrum fragmentation at t_i affects the future arrivals at time $t'_i = t_i + \bar{r}_i, \forall i$. The time domain FI of a link is calculated using Eq. (4.2).

$$\phi_t(l) = \frac{1}{|\mathbb{C}|} \left(\frac{1}{T_M} \sum_{\forall c_i \in \mathbb{C}} \tau_i \right) \tag{4.2}$$

In (4.2), $|\mathbb{C}|$ is the number of connections traversing the link l, and it is used as a normalization factor. T_M is the maximum remaining lifetime of a connection in $\mathbb{C}$. If the remaining lifetime of a connection c_i is $\bar{r}_i$, then $T_M = max(\bar{r}_i), \forall i$. τ_i is the differential remaining lifetime of a connection c_i with respect to the maximum remaining lifetime T_M, i.e. $\tau_i = T_M - \bar{r}_i$.

To explain the time fragmentation metric, consider an example of a fiber link that contains ten slices, and two-dimension (spectrum and time) spectrum occupancy is shown in Fig. 4.1(b). A connection c_5 has the maximum remaining lifetime, i.e., $T_M = \bar{r}_5$. We define the time fragmentation metric as the sum of differential remaining lifetimes of all connections with respect to the maximum remaining lifetime. As shown in Fig. 4.1(b), $\{\tau_1, \tau_2, \tau_3, \tau_4\}$ and $\{0\}$ are differential remaining lifetimes of connections $c_i, i = 1, 2, \ldots, 5$, respectively. Therefore, we can calculate the time fragmentation metric for a link by Eq. (4.2).

The complexity of the fragmentation metric in time domain depends on the number of connections traversing a link. For a given number of links L and slices per link C, the maximum number of connections in the network is LC. Furthermore, the remaining (residual) lifetime of a connection whose holding time is not known in advance depends of the holding time distribution of the connection. Unlike the exponentially distributed holding times, whose residual lifetime is the average value of the holding times, estimation of the remaining lifetime is not straight forward for heavy tailed distributed holding times (for example, [72] considers the estimation of lognormally distributed holding times while allocating and reconfiguring a connection). However, for simplicity, in this paper we assume exponentially distributed holding times of connections. Therefore, the computing complexity of Eq. (4.2) is $O(2L\text{C})$.

4.3.3. Combined Fragmentation Metric

To effectively use the fragmentation index for RSA [36] or triggering defragmentation, we need to consider the fragmentation in spectral, spatial and time domains. As can seen in Fig. 4.1(b), spectrum is fragmented along time axis, and when a connection c_2 terminates after its remaining lifetimes, spectral fragmentation will increase. In that futuristic scenario, we will have three free spectrum blocks of sizes 2, 3 and 1, which results in $\phi_s(l) = 0.73\,(= 1 - \left(\binom{4}{3} + \binom{5}{3} + \binom{3}{3}\right) / \binom{8}{3})$, which is higher than what we have at the moment (i.e., 0.45). Hence, time domain spectrum fragmentation influences the spectral fragmentation as well.

In our approach, we therefore consider a combined fragmentation metric, which takes care of both spectral domain and time domain fragmentation. If $\phi_s(l)$ and $\phi_t(l)$ are the fragmentation metrics in spectral and time domains, respectively, then we define a combined metric named "spectral-time fragmentation metric" (denoted

by ϕ_{s-t}) for a link as follows.

$$\phi_{s-t}(l) = \alpha \times \phi_s(l) + (1 - \alpha) \times \phi_t(l) \tag{4.3}$$

In Eq. (4.3), $\alpha, 0 < \alpha < 1$ is the combining factor, which determines the proportionality in which the two fragmentation metrics to be added. In EONs, link fragmentation constitutes to the overall network fragmentation, and thus we define the network fragmentation index as follows, where L is the total number of fiber links.

$$\phi_{s-t}^{Network} = \frac{1}{L} \sum_{\forall l} \phi_{s-t}(l) \tag{4.4}$$

For the network fragmentation index, the knowledge of fragmentation metric of all links (i.e., spatial dimension) must be gathered at the control plane, and using the above complexity analysis, the computing complexity of the combined metric in a network is given by $O(LC + 2L^2C)$.

4.4. Multi-dimensional Resource Allocation Scheme

Upon arrival of a new request, the control plane tries to provision it the best route and resources (transponder, spectrum, fiber, etc.), if possible, otherwise blocks it when free resources are unavailable and delayed reservation is not accepted to the connection request. Provisioning of resources is done by the network control and management system based on the requirement of connection requests, and it is often policy-driven. However, irrespective of the policies, network operators like to maximize revenue, which is mainly done by serving more requests and fulfilling their needs. In order to propose a fragmentation-aware resource allocation scheme, it is necessary to allocate such spectrum slices that minimizes the spectrum fragmentation in the network. In the following section, we propose a cost function considering spectral, spatial and time domains, and use it for resource allocation in EONs.

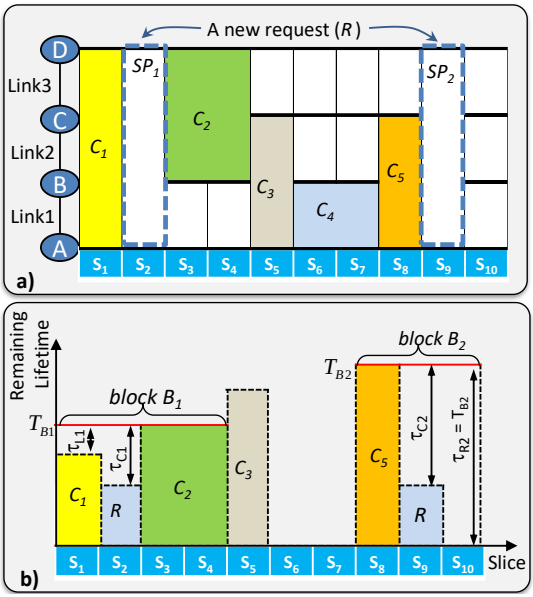

Figure 4.2.: Spectrum occupancy of a 3-link path with 10 slices per fiber-link. (b) Spectrum occupancy of a link 2 along spectral and time domains. This figure shows the provisioning of a request R along other lightpaths taking their remaining lifetimes into consideration in blocks B_1 and B_2 [60].

4.4.1. Holding-Time-Aware RSA

The cost function must take into account the spectrum fragmentation in spectral, spatial and time domains, and using this function the best (minimum cost) spectrum path (SP) could be selected for the resource allocation of a new connection request. We define a spectrum path (SP) by a set of free contiguous and continuous slices which could be allocated to a new request, and a connection is setup

over a SP to transfer client data of one or more connections assigned to it. To illustrate this, consider an example in Fig. 4.2(a), where there are three possible spectrum paths (SPs) in a 3-link linear network, which could satisfy the bandwidth demand of the new request R, assuming that one spectrum slice is sufficient to satisfy its demand. Notice that only two spectrum paths SP_1 over slice s_2, and SP_2 over slice s_9 are shown, and a third SP over slice s_{10} is not shown. Furthermore, to choose the best SP which could minimize spectrum fragmentation in spectral, spatial and time domains, we assign cost to each candidate SP using Eq. (4.5), which takes remaining lifetimes of neighboring lighpaths into account.

$$cost(sp_i) = \sum_{\forall l \in sp_i} (\phi^l(sp_i) + j \times \epsilon), \text{ where}$$

$$\phi^l(sp_i) = \frac{(\tau_{L_i} + \tau_{C_i} + \tau_{R_i})/T_{B_i}}{|B_i|} \quad (4.5)$$

In Eq. (4.5), j is the index of first slice occupied by a spectrum path sp_i, ϵ is a small constant (10^{-3}), and the fragmentation factor ($\phi^l(sp_i)$) of all links (i.e., spatial dimension) traversed by a spectrum path SP_i are taken into account to define the cost of SP_i. The factor $\phi^l(sp_i)$ is calculated as the average value of normalized differential remaining lives of left (τ_{L_1}), center (i.e., incoming, τ_{C_1}) and right ($\tau_{R_1} = 0$) connections (see Fig. 4.2(b)). The normalization constant is the maximum value of remaining lifetimes i.e., $T_{B_i} = \max(\bar{r}_{L_i}, \bar{r}_{C_i}, \bar{r}_{R_i})$ of connections associated with a block B_i. Note that a block B_i in Fig. 4.2(b) is associated with a group of neighboring lighpaths around the possible spectrum allocation of the request R over a candidate spectrum path SP_i.

To visualize these parameters, let us see Fig. 4.2(b), where the spectrum occupancy of link-2 is shown across spectral and time dimensions. Here, two possible spectrum allocation of the new request R are also shown together with other connections in block B_1 and B_2. The differential remaining life of connections in a block

B_i (i.e., τ_{L_i}, τ_{C_i} and τ_{R_i}), are calculated as the difference between the maximum remaining lifetimes of connections in a block B_i and the remaining lifetime of an individual connection, i.e., $\tau_{L_i/C_i/R_i} = T_{B_i} - r_{L_i/C_i/R_i}$. Note that the center connection (actually an SP) in a block B_i refers to a possible spectrum assignment of the new request, and when the spectrum is free to the left or right of a spectrum path SP_i, then τ_{L_i} or τ_{R_i} is considered same as the normalization constant T_{B_i}. This ensures that $cost(sp_i)$ is lower for those SPs that fits into the gap between two connections, as well as the minimum cost SP ensures that incoming request and its neighboring connections have minimum variance of remaining lifetimes, thus they will depart together and eventually will minimize fragmentation in spectral, spatial and time domains.

4.4.2. Dynamic HTI-based Resource Allocation Scheme

We propose a heuristic for dynamic routing and spectrum assignment of incoming requests considering HTs and remaining lives of connections in Algorithm 4. First, on a set of pre-computed fiber-level paths, we find all possible SPs that have sufficient free slices $(b + g)$ to admit a new connection request $R(o, d, b, h)$, where o and d are origin and destination nodes, b is the bandwidth, h is holding time of the request R, when known, otherwise we consider the mean value $E[H]$ as its holding time; and g is the number of guardband slices needed to separate two neighboring lighpaths. We assume that spectrum continuity and contiguity constraint must be satisfied to admit the connection request over a SP. The costs of all candidate SPs are computed in Step 3. Then, a minimum cost path among all SPs in a set $\mathbb{SP}$ is selected to admit the new connection request in step 4.

Algorithm 4 HTA-based Resource Allocation Scheme (HTA-RA) [60]

1: **Given**: Arriving request $R(o, d, b, h)$, network state, g= number of guardbands (in slices).
2: Find all SPs between $o-d$ pair on its fiber-level paths that fulfills the bandwidth requirement, i.e., $b+g$ continuous and contiguous slices.
3: Assign costs to all SPs (in a set $\mathbb{SP}$) as given by Eq. (4.5).
4: Establish a new lightpath over a minimum cost spectrum path to admit the connection request R.
5: Block the request R, if no candidate SP is found.

The complexity of Algorithm 4 depends on the number of candidate LPs and SPs, and computation of their weights. As we pre-compute fiber-level paths for all $o - d$ pairs, and the maximum number of connections in a N-node network with L fiber-links, and C spectrum slices per fiber link is LC, the time complexity for running Dijkstra's κ-shortest paths algorithm to find all SPs is $O(\kappa LC + NC\log(NC))$. Furthermore, the remaining (residual) lifetime of a connection whose holding time is not known in advance depends of the holding time distribution of the connection. For the exponentially distributed holding times, whose residual lifetime is the average value of the holding times, estimation of the remaining lifetimes of LC connections on a maximum of H-hop route can be computed in $O(LCH)$. Therefore, the overall time complexity is polynomial.

4.5. Multi-dimensional Resource Reallocation Schemes

An efficient fragmentation-aware routing and spectrum allocation scheme can alleviate the spectrum fragmentation, however, it cannot be eliminated. The spectrum allocation of heterogeneous demands

with the contiguity and continuity constraints eventually fragments occupied or free spectrum, which needs to be reallocated to keep fragmentation at a minimum level. The connection reconfiguration as a defragmentation method, however, generally results in some form of service disruption. To this end, past efforts have focused on minimizing disruptions during reconfigurations, or their complete elimination [34, 45, 46, 47]. To avoid service disruptions, a connection can be moved from a frequency f_0 to f_3 if the intermediate spectrum (i.e., from f_1 to f_2) on the same route is free. In this way, spectrum fragmentation can be minimized in the so-called *non-disruptive* fashion. However, non-disruptive reconfiguration has its own limits in the extent to which it can minimize the blocking in the network; clearly reconfiguration of the connections can bring more benefits. Hence, the question is whether other methods can increase the resource utilization while minimizing the service disruptions. To illustrate the idea, let us consider an example of spectrum occupation over six slices (or "slot width granularity" in ITU-T G.694.1 [4]) in Fig. 4.3. As it can be seen, in absence of time domain, one of the possible outcomes of a defragmentation method in Fig. 4.3(a) is a perfectly well-utilized spectrum, but only for a short period of time: this is because the short duration connections (c_2 and c_4), once tear down, will fragment the spectrum again. A better method is shown in Fig. 4.3(b). Here, the HTI is used to reconfigure the connections, allowing the system to remain in a non-fragmented state for longer period of time than if no HTI was considered. In the following subsections, we investigate whether holding times can also be exploited for reconfiguring connections in a reactive as well as proactive manners.

4.5.1. HTA-based Reactive Defragmentation Scheme

In a mesh network, both spectrum continuity and contiguity constraints must be considered while (re)allocating spectrum to con-

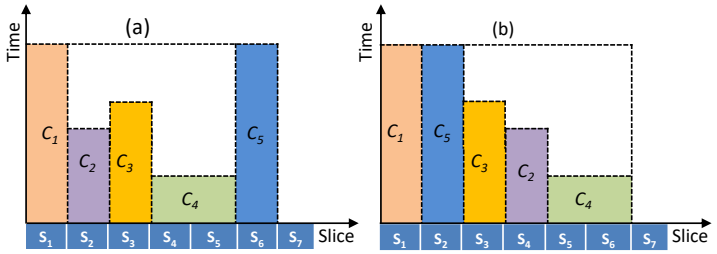

Figure 4.3.: An example of spectrum utilization (a) without holding times information, and (b) with holding times information. Spectrum occupancy along time axis shows the remaining lifetimes of connections [36].

nections, which makes the defragmentation process rather complex. The key idea is that the reallocation of some connections should pave the way for an incoming connection that could not be accepted normally due to the spectrum fragmentation. For example in Fig. 4.4(a), when a new request R between nodes A and D arrives with demand of 2 slices, it can not be admitted into the system normally, since there is not sufficient free block of consecutive spectrum slices. However, when some connections (c_1 and c_2) are moved to other available spectrum slices, then the request R can be admitted into the system.

We adopt an auxiliary graph construction approach [40] to find a solution for reallocating some connections using a maximum independent set (MIS) algorithm proposed in [73]. The MIS algorithm tries to find a new non-overlapping set of spectrum paths (i.e., outside currently occupied spectrum) for connections c_1 and c_2 such that the spectrum slices [2, 3] can be freed for a setting up a connection $c_{(R)}$ for admitting the request R. Here, the connection c_1 (c_2) can be moved to spectrum paths over slice(s) 6 or 7 ([5, 6] or [6, 7]), and these solutions are shown as vertices $c_{1.1}$ and $c_{1.2}$ ($c_{2.1}$ and $c_{2.2}$), re-

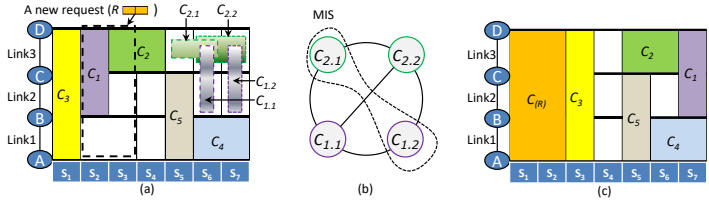

Figure 4.4.: An example of HTA-based reactive defragmentation. a) Possible solutions of reconfiguring connections c_1 and c_2 to accommodate a new request R; b) auxiliary graph constructed for finding MIS; c) Spectrum occupancy after reconfigurations (including the new connection $c_{(R)}$) [60].

spectively as shown in Fig. 4.4(b). Here, two vertices are connected if they represent solutions of same connection or if they spectrally overall. The MIS algorithm then tries to find a possible independent set whose cardinality is same as the number of connections to be reconfigured. Here, a MIS $\{c_{1.2}, c_{2.1}\}$ is found with cardinality 2, therefore the connections c_1 and c_2 can be moved to slice(s) 7 and [5, 6], respectively. Therefore, the spectrum for new request R will allocated at slices [2, 3]. However, since its neighboring connection c_3 also belongs to the same origin-destination A-D pair, connections c_3 and $c_{(R)}$ are (re)allocated to spectrum slice(s) 3 and [1, 2], respectively based on their remaining lifetimes. Fig. 4.4(c) shows the spectrum occupancy of all connections assuming that the remaining lifetime of c_3 is lower than the holding time of the new connection $c_{(R)}$. It is important to notice that the connection c_2 can be reconfigured without disrupting its ongoing traffic using the "push-pull" mechanism [46], on the other hand, the connection c_1 has to crossover a connection c_5, so the traffic of c_1 will be interrupted for a small amount of reconfiguration time, if we use the "make before

break" approach [45].

Algorithm 5 HTA-based Reactive DF Scheme (Reactive-HTA-DF) [60].

1: **Given**: Arriving request $R(o, d, b, h)$, network state, and g: number of guardbands (in slices).
2: Admit a connection request as per a given RSA scheme (e.g., κ-shortest paths routing and FF spectrum allocation).
3: **if** spectrum can't be allocated **then**
4: Reallocate adjacent connections in $\mathbb{C}$ using a non-conflicting solution provided by the MIS algorithm [73] such that b slices can be set free for the request R.
5: **if** connection request R can be established **then**
6: Rearrange connections in $\mathbb{C}$ in decreasing order of their remaining lifetimes.
7: **else**
8: Reallocate all connections in spectrum interval $[i, i+b+g-1]$ for each possible index i, $1 \leq i \leq \mathrm{C} - b - g + 1$, where C is total number of slices, until R is established.
9: **end if**
10: **end if**
11: If SA in not possible for the request R, block this request.

We propose a heuristic for dynamic Hooding-Time-Aware (HTA)-based reactive reconfiguration scheme in Algorithm 5. When a connection request $R(o, d, b, h)$ arrives then we try to allocate it as per a given RSA policy, e.g., κ-shortest paths routing and FF spectrum allocation or the HTA-RA scheme described in Algorithm 4 (in step 2). If there is no free spectrum that can be assigned to the request R (step 3), then we find all other $o - d$ connections, and try to move their adjacent connections using a MIS algorithm such that the request can be allocated together with other $o - d$ connections,

in a decreasing order of their lifetimes, with additional b subcarrier slices (steps 4-6). If the request can not be grouped with other $o-d$ connections, then we try to reallocate other connections to set free $b+g$ consecutive spectrum along its shortest path, and the process is repeated for all possible index i until the request is allocated (steps 7-8). Finally, the request R is blocked if spectrum can not be allocated even after the defragmentation process.

As described above, the Algorithm 5 mainly consists of four parts: i) finding the desired first shortest optical path; ii) calculating network fragmentation metric; iii) finding a non-conflicting solution using an MIS algorithm; and iv) reallocation of spectrum to some connections. As we pre-compute fiber-level paths for all $o-d$ pairs, and the maximum number of connections in a N-node network with L fiber-links and C spectrum slices per fiber-link is LC, the time complexity for running Dijkstra's κ-shortest paths algorithm is $O(\kappa LC + NC\log(NC))$. As mentioned in the previous section, the network fragmentation metric can be calculated in the order $O(LC + 2L^2C)$. We preferred the algorithm in [73] as it can achieve a better complexity of $O(2^{0.0076N})$ to find the MIS and possibly a non-conflicting reallocation solution. Finally, the rearrangement of the maximum possible connections based on their remaining lifetimes (assuming exponentially distributed holding time distribution) can be accomplished in $\sim O(LC^2)$ computation time. The overall complexity for running the Algorithm 5 is the sum of the computation times of all major parts, and for a large network, it can be approximated as $\sim O(LC \times (C + \log(NC) + 2^{0.0076N}))$. So, the time complexity of the reactive reconfiguration scheme is non-polynomial.

4.5.2. HTA-based Proactive Defragmentation Scheme

For the purpose of non-disruptive reconfiguration, we now introduce a HTA-based proactive defragmentation scheme that reconfigures some connections without crossing-over the other connections in the

system. Let us consider an example of a three-links fiber path in Fig. 4.5(a), where in addition to guardband occupancy, spectrum allocation is shown for the existing connections. When a new con-

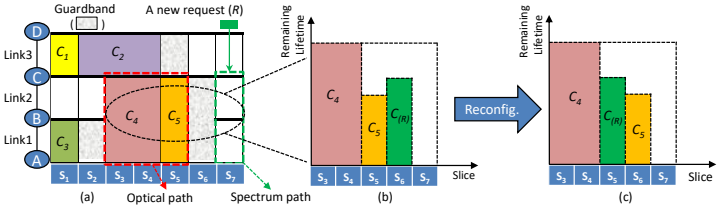

Figure 4.5.: An example of reconfiguration based on the holding times. a) Occupancy of links when a new request R arrives; b) the time-frequency view of connections' occupancy over slices s_3 to s_7, if $c_{(R)}$ is established at slice s_6; c) request c_5 is reconfigured to allow SA for $c_{(R)}$ at slice s_5 [60].

nection request R arrives between origin A and destination C, it can be allocated over spectrum path at slice 7. However, our dynamic proactive reconfiguration scheme selects a slice 6 first to establish R together with other connections associated with an optical path of A-C pair, and shifts the guardband to slice 7. The occupancy of c_4, c_5 and R in frequency and time domains can be seen in the Fig. 4.5(b). Here, R has higher remaining lifetime (shown by its length along time axis) than its neighbor c_5. If we do not reconfigure c_5 from spectrum slice 5 to 6, then termination of the connection c_6 would fragment the spectrum on links 1 and 2. This shows that the fragmentation in spectral and spatial domains is highly dependent on the holding times of the requests, and its time domain behavior. Therefore, we reconfigure the connection c_5 from slice 5 to 6 without affecting other requests in the system, and establish the connection request R over slice 5 (see Fig. 4.5(c)). Let us assume that

reconfiguration time from slice i to j is in $|i-j|$ time units. Therefore, reconfiguration can be implemented in $O(\mathbb{C}| \times \mathrm{C})$ computation time [34], where $|\mathbb{C}|$ is the number of connections to be reconfigured, and C is the total number of spectrum slices. In a more complex scenario, where spectrum slices of different links are occupied by different $o-d$ flows (connections), non-disruptive reconfiguration is likely much more complex. Therefore, in proactive defragmentation scheme, we focus on reconfiguring only such connections, which can be moved without crossing over the spectrum of other connections. To this end, we try to allocate spectrum for an incoming request between a $o-d$ pair along other connections of $o-d$ pair, and reconfigure some of its connections based on their remaining lifetimes in order to minimize the spectral and spatial fragmentation.

Algorithm 6 HTA-based Proactive DF Scheme (Proactive-HTA-DF) [60]

1: **Given**: Arriving connection request $R(o,d,b,h)$, network state, and g: number of guardbands (in slices).
2: Find a first shortest optical path between $o-d$ pair which can be expanded to admit C with b additional subcarrier slices along other connections $\mathbb{C}$ of $o-d$ pair.
3: **if** C can be admitted along with other connections $\mathbb{C}$ of $o-d$ pair **then**
4: Get the remaining lifetimes ($\bar{r}_i$) of connections in $\mathbb{C}$ and establish request C with other connections in $\mathbb{C}$ in a decreasing order of their remaining lifetimes.
5: **else**
6: Establish it over first available spectrum path with $b+g$ subcarrier slices (or, use other SA policy).
7: **end if**
8: If SA in not possible for the request R, block this request.

Algorithm 6 describes our proactive heuristic for spectrum assignment and reconfiguration based on the holding times of connections. When a new request $R(o, d, b, h)$ arrives, we try to route it on shortest path between nodes s and d, and allocate spectrum along other connections between $o-d$ pair, otherwise establish it over first available spectrum path (similar to Algorithm 5). Holding times information based proactive reconfiguration plays a role when the request R can be established along other connections of $o-d$ pair (step 3). All connections (including R) of $o-d$ pair are reconfigured (allocated) based on decreasing order of their remaining lifetimes (step 4). The time complexity analysis of the Proactive reconfiguration scheme can be done in similar way as reactive scheme, and it can be given as $\sim O(LC \times (C + \log(NC))$, which is polynomial.

4.5.3. Performance Evaluation

We now evaluate the performance of our HTA-based resource allocation scheme (HTA-RA), HTA-based reactive DF scheme (Reactive-HTA-DF), and HTA-based proactive DF scheme (Proactive-HTA-DF) in terms of bandwidth blocking ratio (BBR), spectrum utilization (SU), and the number of reconfigurations performed per defragmentation. We compare the BBR of our HTA-based DF schemes against the regular system without reconfiguration under the FF spectrum allocation policy, and also against the spectrum defragmentation without HTA using [40]. We simulate the heuristics on the 14-node NSFNET [Fig. 2.10(a)] and the 15-node Atlanta [Fig. 2.10(b)] topologies, and use shortest fiber-level path for routing and spectrum assignment. Each fiber-link is bidirectional (consists of two unidirectional fibers) and each fiber has a total spectrum width of 2.5 TeraHertz. We assume that a subcarrier slice occupies 12.5 GHz, thus, there are 200 spectrum slices on each fiber link. We evaluate the proposed schemes under different number of guardbands (g=1 and 2), which is kept on either (or both) side(s) of an opti-

cal path that carries one or more connections belonging to the same origin-destination ($o-d$) pair to avoid the cross-channel interference between different $o-d$ connections (In a real system, OFDM carriers might not be orthogonal completely, so guard bands are necessary to reduce cross-talk and to improve security). We assume that a spectrum slice can carry requests with a total of 50 Gb/s using DP-QPSK modulated OFDM technology.

In the simulation, we consider three types of requests with demands 40 Gb/s, 100 Gb/s and 400 Gb/s, which require 1, 2 and 8 consecutive slices, respectively. We randomly generate the requests for each uniformly selected origin-destination node pair. The arrival rates of requests are proportional to the number of ports, which is in the ratio of 8 : 4 : 1 for 40, 100 and 400 Gb/s, respectively. The holding times of the requests are exponentially distributed with mean service time as 10 time units. If the holding time is not known in advance for an incoming connection request, then we use its mean service (holding) time. The combining factor $\alpha = 0.5$ is used to assign equal weight to spectral and time fragmentation metrics in Eq. (4.3) and (4.4) while calculating the network fragmentation index ($\phi_{s-t}^{Network}$), which is used for triggering reactive defragmentation in Algorithm 5. Unless otherwise stated, we assume that the holding times of all connections are known. The traffic load is varied from 300 to 800 Erlangs. We generate 1,000,000 requests to evaluate the bandwidth blocking ratio, defined as the ratio of blocked bandwidth demands to the total offered bandwidth demands of all requests.

In order to answer when to best use holding time information, i.e., during resource allocation or during resource reallocation or during both the phases, in Fig. 4.6, we first evaluate the BBR performance of FF spectrum allocation policy and HTA-based [36] resource allocation scheme (HTA-RA), and apply our DF (Proactive and Reactive) methods on top of both resource allocation policies. The FF policy assigns a first available (lowest index) spectrum path to the

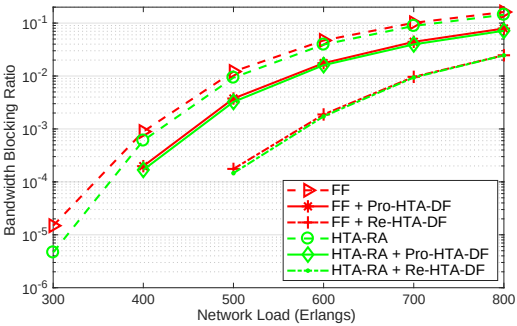

Figure 4.6.: Bandwidth blocking ratios of the FF and the HTA-RA schemes, and defragmentation methods applied on top of both RA policies with number of guardbands $g = 1$, and DF threshold $\gamma_{th} = 0.3$.

new request. The HTA-RA scheme uses the remaining lifetimes of connections during the admission of a new connection request, as explained in Algorithm 4. As we can see, the HTA-RA scheme has lower blocking than FF policy, which proves our hypothesis of utilizing holding times in fragmentation-aware routing and spectrum allocation. When we apply our HTA-based DF schemes, then BBR reduces in both cases, and the BBR reduction is significant for DF with a reactive approach. However, both the FF and HTA-RA policies with reactive DF provide the similar BBRs. Therefore, we can say that it is better to use a simple yet very effective FF spectrum allocation policy than complex RSA schemes such as HTA-RA, and apply an efficient HTA-DF scheme which can reduce the BBR significantly with minimum traffic interruptions. In the remaining performance evaluation scenarios, we apply the FF for spectrum allocation and apply our DF methods.

Next, we evaluate the BBR performance of both reactive and

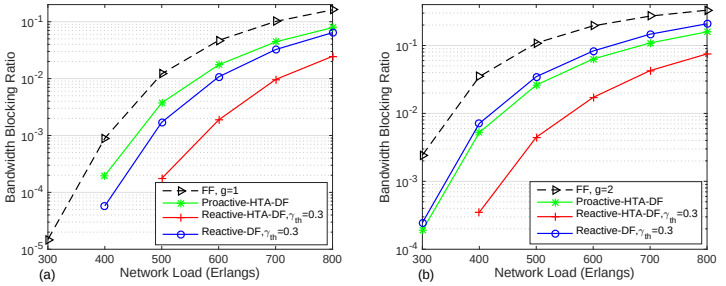

Figure 4.7.: Bandwidth blocking ratios of FF policy, and defragmentation methods with and without HTA: (a) number of guardbands $g = 1$ and (b) number of guardbands $g = 2$ (NSFNET) [60].

proactive HTA-based DF schemes using Algorithm 5 and 6, respectively for two different number of guardbands ($g = 1$ and $g = 2$) and complete knowledge of holding times of connections. We compare the BBRs of both schemes with the FF spectrum allocation scheme which does not perform defragmentation, and with a Reactive-DF [40], which performs the reactive defragmentation on top of the FF spectrum allocation policy, but without the knowledge of holding times of connections. For both the Reactive-HTA-DF and Reactive-DF schemes, fragmentation threshold is set to a lower value ($\gamma_{th} = 0.3$), such that if a connection could not be accepted normally then DF would be triggered, i.e., $\phi_{s-t}^{Network} > \gamma_{th}$.

Fig. 4.7(a) shows the BBR in a regular system (i.e., without DF) under FF spectrum allocation policy and DF schemes with one guardband. As expected, defragmentation schemes reduce blocking by at least an order of magnitude than the regular system. On comparing the HTA-based DF schemes (proactive and reactive) with a spectrum DF without HTA (Reactive-DF), what we

observe is that both reactive DF schemes with and without HTA outperforms the Proactive-HTA-DF scheme, which is due to the fact that the Proactive-HTA-DF allows only non-disruptive reconfiguration and restricts reconfigurations between same origin-destination pair. Reactive-DF schemes cause traffic disruption, since they explore all possibility of connection reconfigurations, including crossing over spectrum of other connections. And, in fact, the Reactive-HTA-DF scheme outperforms all other schemes, which shows the effectiveness of the knowledge of holding times. Interestingly, when number of guradband increases, the Proactive-HTA-DF scheme exhibits lower blocking than the Reactive-DF in Fig. 4.7(b). The reason is that spectrum occupancy due to guard band increases more in Reactive-DF scheme than the Proactive-HTA-DF scheme, since Reactive-DF is performed in a regular system with the FF spectrum allocation which establishes each connections with the guardband g, the Proactive-HTA-DF scheme, on the other hand, allows to group a new arrival with other $o - d$ connections. We observe the similar BBR performance improvement in the Atlanta network topology when the proactive and reactive HTI-DF are applied [60].

In Fig. 4.8, we plot the average spectrum utilization gain (in percent) due to reconfiguration in the Proactive-HTA-DF and Reactive-HTA-DF schemes, which is calculated as the percentage change in the spectrum utilization of regular system with FF policy when defragmentation is applied. Note that we calculate the spectrum utilization without taking into account the guardbands, and we get the average value by averaging SU over all the 42 links of the NSFNET. As we can see that reconfiguration of connections improves the spectrum utilization for both DF methods, and it increases with the load values. However, the reactive method is the more beneficial, and the spectrum utilization gain is more than 20% at a higher load value of 800 Erlangs, as it accepts relatively more connections.

Fig. 4.9 shows the average number of connections whose spectrum

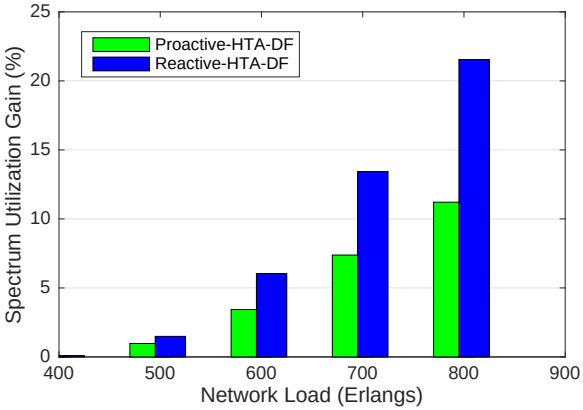

Figure 4.8.: Spectrum utilization gain (in percent) due to defragmentation as compare to the regular system when no defragmentation is applied (NSFNET) [60].

paths are reconfigured for every time a defragmentation is performed in NSFNET. As we can see that the average number of reconfigurations in the Reactive-HTA-DF method is relatively higher, since it allows some of the connections to cross-over spectrum of other, which is disruptive in nature, in addition to the non-disruptive way of shifting of connections during the reconfiguration process. However, more number of reconfigurations also result in lower connection blocking, see Fig. 4.7. The number of reconfigurations increases up to a moderate load (400 Erlangs), but at higher loads it reduces again, since spectrum is mostly occupied at higher loads. The Proactive-HTA-DF method, on the other hand, allows only reconfiguration of connections between the same origin-destination $(o - d)$ node-pair when a new arrival between $o - d$ is accepted into the system. Note that, proactive method does not disrupt the ongoing traffic between any connections during reconfiguration. Moreover,

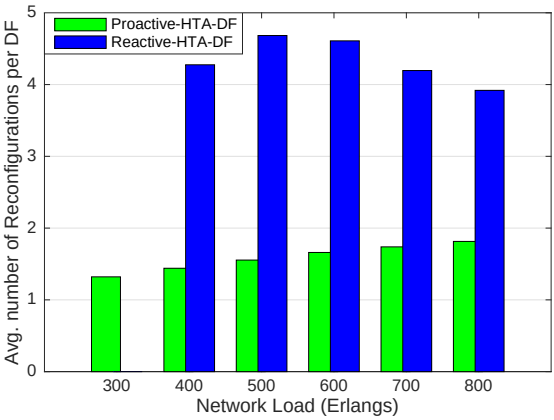

Figure 4.9.: Average number of reconfigured connections per DF [60].

the number of reconfigurations (slightly) increases with the increase in load. The reason is that the as load (or arrival rate) increases, more number of connections between a $o-d$ node-pair try to rearrange themselves based on their remaining lifetimes while admitting a connection between the $o-d$ pair.

4.6. Summary

This chapter outlined the spectrum fragmentation issue in elastic optical networks, and presented novel solutions, including a proactive-non-disruptive defragmentation scheme and a reactive-disruptive defragmentation scheme in addition to a fragmentation-aware resource allocation scheme, which utilized the holding time information of connections.

In contrast to the conventional first-fit policy for allocating first available resources to a new connection request, we focused on spec-

trum fragmentation in spectral, spatial and time domains to design a novel holding-time-aware resource allocation scheme. We showed that the proposed scheme can outperform the FF allocation method at the cost of higher complexity. Additionally, we utilized the holding time information during the resource reallocation phase of a proactive defragmentation scheme and a reactive defragmentation scheme. We showed that the HTA-based DF schemes are very effective in reducing blocking and thus increasing spectrum utilization in the network. The gains with these methods come due to the reconfiguration of some connections. While the proactive DF method reconfigures connections in a non-disruptive manner, the reactive method, in contrast, allows crossing-over between connections during reconfiguration, hence disruptive in nature, however results in lower blocking and higher spectrum utilization gain as compare to proactive DF method. With a smaller number of reconfigurations and the knowledge of holding times of connections, network operator can reduce connection blocking significantly and increase its revenue by increasing spectrum utilization.

5
Application of Machine Learning in Resource Provisioning

5.1. Introduction

As a scientific discipline, Artificial Intelligence (AI) encompasses very diverse areas including search algorithms and optimization theory, metaheuristics, game theory, knowledge-based reasoning and planning methods, statistical models and decision-making algorithms, as well as learning methods, most notably, Machine Learning (ML) [74]. Many of these techniques have been used in the optical networking arena for a long time, but their popularity, especially machine learning, is gaining momentum [75,76,77,78,79]. Fig. 5.1 shows the uses of AI in optical transmission and networking. The AI indeed provides ample opportunities for automating operations and introducing intelligent decision making in network control and management, while addressing issues like resource provisioning, self-configuration and self-optimization, performance monitoring, through monitoring, prediction and estimation by utilizing present network state and historical data. Especially, statistical and probabilistic learning methods like Bayesian learning, hidden Markov models, Kalman filtering, and ML techniques like case-based reasoning, support vector machines, neural networks and random forests have been applied efficiently to perform some of these tasks. ML can also help improve the configuration and operation of network

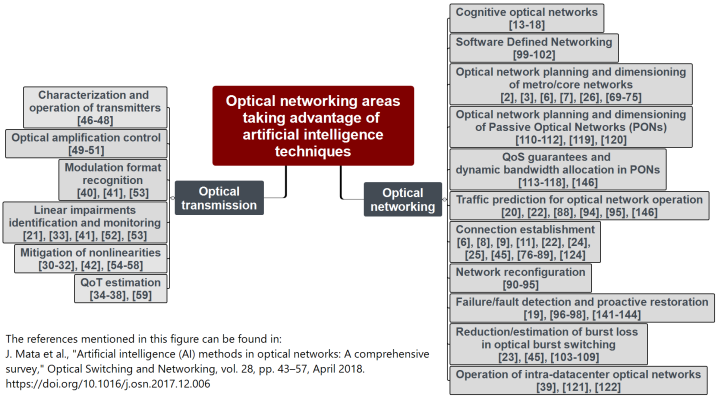

Figure 5.1.: Various use cases of AI in optical communication [78]. The references shown in the figure correspond to those presented in [75].

devices, optical performance monitoring, modulation format recognition, fiber nonlinearities mitigation and Quality of Transmission (QoT) estimation, resulting in overall better network performance. AI, in general, can make the optical networks cognitive [80]. Cognitive optical networks can be supported by intelligent controller functions implemented in Software Defined Networking (SDN) framework. The intelligent cognitive controller exploits the data collected from network monitors and additional data sources with the use of AI techniques to improve network performance.

We saw in the previous chapters that spectrum fragmentation is a critical issue in resource (re)allocation in optical networks. Moreover, in inter- and intra- datacenters, resource provisioning is a big challenge due to diverse running applications and associated time-varying traffic of intra- and inter- datacenter communication, which require different amount of latencies and resources. Therefore, the question is can resources in optical networks in general or in data-

centers be provisioned efficiently and intelligently utilizing learning techniques? In resources allocation for optical datacenter networks the best scenario is for the applications to announce their session sizes (holding times) at their requests for network resources. This thesis treats the case when such prior handshake is not available and ML techniques are used to predict holding times of services. Since exponential holding times do not provide any predictions of residual life but is necessary for the Markov model, the thesis refines its treatment to other holding time distributions such as log-normal, so residual life can be predicted and used to optimize allocations and reallocations. Machine learning techniques, especially deep recurrent and convolutional neural networks are shown to be beneficial in learning repeated patterns. Specially, for sequential time-series patterns, such as traffic-variation throughout day or month, recurrent neural networks are proved to be quite useful. But of course there is no proof of optimality or even convergence as in most ML cases. In case of black swan events, or rare situation which have not been seen before, prediction or classification is even more challenging. Furthermore, cloud computing paradigm is changing to distributed computing, where in addition to compute, process and transmission related challenges, energy consumption is a big issue. Another important issue is optical layer security, since optical devices are prone to eavesdropping and jamming attacks [81]. AI techniques can be helpful in mitigating some of these emerging challenges, and the subsequent sections in this chapter tries to address some of these challenges through ML techniques. However, ML also needs to deal with challenges due to the complexity of its emerging learning models, particularly deep learning, and more importantly, in handling large sample of collected data through heterogeneous sensors and network devices, data breaches, and time-varying parameters.

The remaining sections of this chapter present two specific applications of machine learning techniques for resource management in op-

tical networks. Among the two specific use cases, first we present resource (re)allocation schemes in optical datacenters for time-varying traffic with the help of machine-learning-based traffic prediction approach. The second use case highlights the energy and bandwidth management in fiber-wireless networks for animal-human cohabitation.

5.2. Supporting Publications

- J. Mata, I. de Miguel, R. J Durán, and N. Merayo, S. K Singh, A. Jukan, and M. Chamania, "Artificial intelligence (AI) methods in optical networks: A comprehensive survey," *Optical Switching and Networking*, vol. 28, pp.43-57, 2018. `https://doi.org/10.1016/j.osn.2017.12.006`

- S. K. Singh and A. Jukan, "Machine Learning-based Prediction for Resource (Re)allocation in Optical Data Center Networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, issue 10, pp. D12-D28, 2018. `https://doi.org/10.1364/JOCN.10.000D12`.

- S. K. Singh, F. Carpio, and A. Jukan, "Improving Animal-Human Cohabitation with Machine Learning in Fiber-Wireless Networks," *MDPI Journal of Sensor and Actuator Networks*, Special Issue on Optical Technologies for IoT, Smart Industry and Smart Transportation, vol. 7, no. 3, 2018. `https://doi.org/10.3390/jsan7030035`.

- J. Mata, I. de Miguel, R. J Durán, and N. Merayo, S. K Singh, A. Jukan, and M. Chamania, "Application of Artificial intelligence Techniques in Optical Networks," *IEEE Photonics Society Summer Topical Meeting Series (SUM)*, Hawaii, USA, July 2018.

5.3. Handling Time-Varying Traffic in Optical Datacenter Networks

Intra- and inter- Optical Data Center Network (ODCN)s are emerging as a viable solution to cope up with the increase in data center network (DCN) traffic and also to mitigate the power consumption of its counterpart, i.e., electrical DCNs. The DC traffic can be categorized into short-, medium- and long-lived based on the flow completion time, which ranges from few milliseconds to hundreds of seconds. Majority (more than 80%) of the DC traffic flows are mice or short-lived flows that are mostly latency-sensitive, however more than 80% of the entire traffic volume is carried by long-lived or elephant flows [82, 83], which require high throughput and they are mostly latency-insensitive and deadline-agnostic (e.g., virtual machine migration), see Table 5.1. In another study – limited to Facebook's services [84], authors found that many flows are long lived but not very heavy, and traffic remains stable over even sub-second intervals. On the other hand, inter-DC traffic can be classified into two categories [85]: *background* DC-to-DC (e.g., data replication), or *client-triggered* (e.g., video streaming), which exhibit varying trends over the day but could be predicted using learning methods [49]. Thus, given the DCN traffic forecast and acknowledging its periodicity, there is a need for applying learning algorithms for better resource utilization both in ODCNs as well as in servers.

Among many AI-based technologies, machine learning (ML) is the most preferable technique for learning, classification/identification and prediction. Many variants of brain-inspired neural networks, such as deep/recurrent/convolutional neural networks, are often used as ML tools that learn from input sets and from the past in order to classify/identify test objects and predict the future [75]. As modern applications, especially cloud-based, are heterogeneous and diverse in nature, holding (service) time information (HTI) of

applications is not known in advance, and they are heavy-tailed distributed [82,83]. Furthermore, intra- and inter-datacenter traffic follows a repetitive pattern during a week or a month. Thus, ML-based techniques can be handy in predicting or forecasting future traffic– in terms of mean arrival rate and mean HTI of traffic flows, and other network related issues like connection blocking. Although the prediction of mean HTI is useful in admission control, it is not sufficient. The reason is that when a new service request arrives, then it could be established with other flows having similar remaining (residual) lives such that the termination of these neighboring flows leads to a reduced spectrum fragmentation [72]. At the same time, the residual life of an active (heavy-tailed distributed) flow or connection is a function of already spent time (i.e., active lifetime) and its mean HTI. Thus, in absence of known mean HTI, its prediction becomes necessary to utilize the mean residual life (MRL) information. Furthermore, the DC traffic exhibits heavy-tailed characteristic, i.e., holding times of some flows are quite large, which make most of the existing works [42,86,87,88,89] inapplicable for resource (re)allocation in ODCNs under the time-varying traffic, since they assume exponentially distributed HTI model, thus have not utilized MRLs of active connections in handling resources in ODCNs.

Table 5.1.: Diversity of intra- and inter-data center applications [90].

	Application examples	Short-lived (milliseconds)	Long-lived (seconds)	Latency-sensitive	Latency-insensitive	Deadline-aware	Deadline-unaware
Intra-DC	Web search, IM, SR, haptics	✓		✓		✓	
	Data mining, storage, analytics		✓		✓		✓
	Email, web services, robotics	✓		✓		✓	✓
	Back-ups, downloads, test-beds		✓		✓		✓
	YouTube, games, VR, smart city	✓	✓	✓	✓	✓	✓
Inter-DC	VM migration, data replication, LB		✓		✓		✓
	Search, file sharing, video streaming	✓	✓	✓	✓	✓	✓

IM: instant messaging; SR: sensor readings; VR: virtual reality;
VM: virtual machine; LB: load balancing.

To address the above issue, in this section we use a ML model to predict time-varying mean HTI of three classes of services– distin-

guished by their bandwidth demands and mean service times (HTI), and use them to estimate MRLs of active flows whose holding times are assumed to be unknown and lognormally (heavy-tailed) distributed. However, the question is how to best use the predicted and estimated traffic-related information in (re)allocation of resources (transponders, fiber-cores and spectrum) in ODCNs? To answer this question, we propose a novel ML-based resource allocation and reallocation scheme in ODCNs that uses the predicted mean HTI, MRLs of active flows, and known HTI of departed connections (note that traffic flow and connection are used interchangeably) to allocate resources to a new connection request. More precisely, we propose cost metrics that utilize the predicted and estimated traffic information as well as requirements of a new connection request to either provision it over an existing lightpath by means of traffic aggregation or route it over a new lightpath. Additionally, the knowledge of past and predicted blocking information is used to trigger an on-demand defragmentation, which is essential in reducing spectrum fragmentation in multi-core fiber links used as interconnects in ODCNs.

5.3.1. Estimation of Mean Residual Life of a Connection

Generally, there are two kinds of application requests– one with *known* HTI and other with *unknown* HTI. Let us assume that a connection request j with *known* HTI advertises its value h_j at the start of service t_{S_j} as shown in Fig. 5.2. For example, a fixed size data volume has to be transmitted at a given interface speed, or a video streamed at fixed bandwidth is active for a predefined time. Thus, at any time t_c during the service of the request j, we know its remaining life is $r_j = h_j - (t_c - t_{S_j}) = h_j - d_j$, where we define d_j as its history. The other type of applications do not advertise their HTI (*unknown* HTI). In this case we assume a random value of the HT, h'_j, but drawn from a known probability distribution $F_H\{t\} = P\{H \leq t\}$, which is used to derive its mean value

$E\{H\}$ and the distribution of its residual life $r_j = h'_j - d_j$, which is given by $F_{\mathcal{R}}(t, d_j) = P\{H \leq d_j + t \mid H > d_j\}$, where a continuous and positive random variable $\mathcal{R}$ represents the residual life. Notice that the residual life distribution depends on the connections' history d_j. The mean residual life (MRL) of a connection j with history d_j is defined as $\bar{r_j}(d_j) = \int_0^\infty (1 - F_{\mathcal{R}}(t, d_j))\, dt$, or without calculating the residual life distribution, the MRL value $\bar{r_j}(d_j)$ is given by $E[H - d_j \mid H > d_j] = \left[\int_{d_j}^\infty (F_H^c(t))\, dt\right] / F_H^c(d_j)$, where $F_H^c(t) = 1 - F_H(t)$ is the so-called survival function [91].

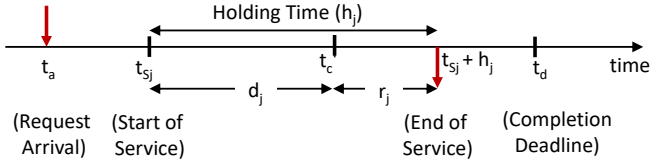

Figure 5.2.: Various traffic parameters of an application request [90].

As requests with known holding times have been well-investigated, our focus is on applications whose lifetimes are mostly unknown but derived either from a memoryless exponential distribution with mean $E[H]$ or from a more practical heavy-tailed distribution like lognormal [82, 83] function with parameters μ and σ. Due to the memoryless property of the exponential distribution, the MRL of a connection is same as its mean holding time $E[H]$. On the other hand, lognormal distribution requires past knowledge, i.e., how long the connection has been already served. Lognormal cumulative distribution function (cdf) is given by $F_H(t) = \frac{1}{2}\left[1 + erf\left(\frac{\ln t - \mu}{\sigma\sqrt{2}}\right)\right], t > 0, \sigma > 0$, and the corresponding probability distribution function (pdf) is $f_H(t) = exp[-\left(\frac{\ln t - \mu}{\sigma\sqrt{2}}\right)^2]/t\sqrt{(2\pi\sigma^2)}$. Here, the error function $erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt$. The mean lifetime is $E[H] = e^{\mu + \sigma^2/2}$, and coefficient of variation (CoV) is $c_v = \sqrt{e^{\sigma^2} - 1}$. In general, for

a heavy-tailed distribution function like lognormal which does not has closed form residual distribution, calculating MRL requires double integrals to be solved by some approximations (e.g., trapezoidal rule) which are mostly time consuming and error-prone using computer simulation. Thus, as suggested in [91, 92], the MRL value for any distribution, including lognormal, satisfying $t \times F_H^c(t) \to 0$ as $t \to \infty$ can alternatively be calculated as

$$\bar{r_j}(d_j) = \frac{1}{F_H^c(d_j)} \int_{d_j}^{\infty} t f_H(t) dt - d_j \tag{5.1}$$

where $f_H(t)$ is the pdf of holding time H. The value of $\int_{d_j}^{\infty} t f_H(t) dt$ for lognormally distributed variable H is given as below, after substituting $y = \frac{\ln t - \mu}{\sigma\sqrt{2}}$ and $y(d_j) = y(t = d_j)$.

$$\begin{aligned}
\int_{d_j}^{\infty} t f_H(t) dt &= \frac{1}{\sqrt{\pi}} \int_{y(d_j)}^{\infty} exp[-y^2 + y\sigma\sqrt{2} + \mu] dy \\
&= \frac{exp[\mu + \sigma^2]}{2} \times \frac{2}{\sqrt{\pi}} \int_{y(d_j)}^{\infty} exp[-(y - \frac{\sigma}{\sqrt{2}})^2] dy \\
&= \frac{E[H]}{2} \times [1 - erf\left(\frac{\ln d_j - (\mu + \sigma^2)}{\sigma\sqrt{2}}\right)]
\end{aligned}$$

Finally, using the value of above term in Eq. (5.1), the MRL of a lognormally distributed holding time of a request j with history (passed time) d_j is given by Eq. (5.2).

$$\bar{r_j}(d_j) = E[H] \times \frac{1 - erf(\frac{\ln d_j - (\mu + \sigma^2)}{\sigma\sqrt{2}})}{1 - erf(\frac{\ln d_j - \mu}{\sigma\sqrt{2}})} - d_j \tag{5.2}$$

The dependence of MRL $\bar{r_j}(d_j)$ on mean holding time $E[H]$, history d_j and CoV c_v is illustrated in Fig. 5.3, where the numerical (theoretical) value of Eq. (5.2) is verified with a Matlab simulation. As can be seen that the relationship between the MRL and the passed time (history) is non-linear and depends on the mean

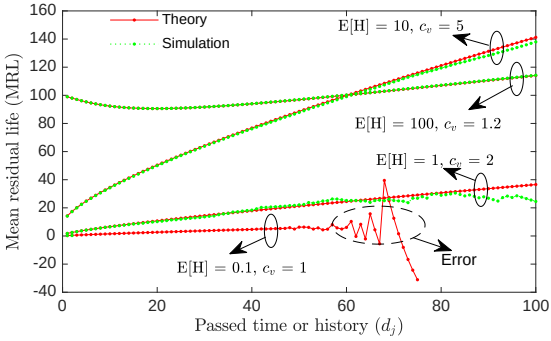

Figure 5.3.: Comparison of MRLs obtained by Eq. (5.2) and simulation for various mean holding time $E[H]$ and CoV (c_v) for lognormal distribution [90].

$E[H]$ and CoV c_v (or equivalently μ and σ). The heavy-tailed nature of lognormal distribution can also be seen even for smaller $E[H] = \{1, 10\}$, since MRL does not die with increasing passed time. It should be noted that when mean holding time is very small $0 < E[H] < 1$, computing Eq. (5.2) could produce wrong MRL values for higher passed times. Furthermore, the simulation for the scenario $E[H] = 0.1$ and $c_v = 1$ does not generate lognormally distributed random variables with higher lifetimes, thus the MRL obtained by the simulation is only shown (in green) for lower values of passed times.

The remaining lifetimes of a lightpath can precisely be calculated by finding mean value of the residual life distribution, i.e., $max(\mathcal{R}_1, \mathcal{R}_2, \cdots, \mathcal{R}_n)$. But the solution requires a double integral to be solved, which is complex and time consuming. Thus, using the knowledge of remaining lifetime, in case of connections with known HTI, or an estimated mean residual life (MRL) of connections with unknown HTI, the remaining life of a lightpath can be estimate

as $r_{lp_i} = \max(\bar{r_1}(d_1), \cdots, \bar{r_n}(d_n))$, where the lightpath is assumed to serve n connections with unknown lifetimes $H_1, H_2, \cdots, H_n$, but known history (passed times) $\vec{d} = \{d_1, d_2, \cdots, d_n\}$. When connections associated with a lightpath have known holding times, then the remaining lifetime of the lightpath is simply given as the $r_{lp_i} = \max(r_1, \cdots, r_n)$, where the remaining lifetime of an associated connection is $r_i = h_i - d_i$.

HTI and MRL are not only useful in provisioning of new lightpaths in EONs, as shown in the example in Fig. 4.2 and Eq. 4.5 in the previous chapter, it can also be used to assigning an already established lightpath to carry the aggregated data, including the traffic of a new connection request. Let us explore this scenario in the next subsection.

5.3.2. Aggregation of Traffic Flows

Aggregation of traffic flows at electrical layer is the easiest solution to accommodate a new traffic flow request. Thus, it is advisable to route a class-k request R along an existing not-fully utilized lightpath, if the lightpath's residual (free) bandwidth is sufficient to satisfy the request's demand (b_k Gb/s), between the same source-destination pair in order to reduce resource consumption and connection blocking. Let us assume that a lightpath LP_1 carries some traffic flows (say, $A_1, A_2, \cdots, A_n$) with aggregated bandwidth demand of B Gb/s, and each spectrum slice can support a maximum of C Gb/s with a given modulation format. Therefore, the lightpath LP_1 can accommodate the traffic flow request R easily at the electrical layer if $B + b_k \leq C$. In other words, the residual bandwidth of the lightpath LP_1 after allocating the demand b_k is non-negative, i.e., $rb(lp_1) = C - B - b_k \geq 0$. Otherwise, there could be another option to expand the bandwidth of the lightpath LP_1 by assigning an additional OFDM subcarrier to the corresponding transponder, if an adjacent slice is free. The flow request R could be carried over

the expanded lightpath LP_1 if $B + b_k \leq 2C$. At the same time, we assume that transponder's capacity is two or more slices. Nevertheless, when there are more candidate lightpaths that could provision a request, then it is wise to devise the knowledge of their residual (free) bandwidth to select the best one, and the cost function is given by Eq. (5.3).

$$cost(lp_i) = p_i \times \left[\left(1 - \hat{rb}(lp_i)\right) \times \epsilon + \frac{|\bar{h}_k^e - \bar{h}_k^p|}{max(\bar{h}_k^e, \bar{h}_k^p)}\right] \tag{5.3}$$

The cost of a candidate lightpath, which could accommodate a new request R, defined in Eq. (5.3) has two factors apart from a multiplying hop-length factor (p_i) and a constant $\epsilon = 10^{-1}$. The first factor $(1 - \hat{rb}(lp_i))$ assigns lower cost to a lightpath that has more normalized residual bandwidth ($\hat{rb}(lp_i)$), which is normalized w.r.t. to the lightpath's bandwidth. Thus, aggregation of requests over a not-fully utilized lightpath gets lower cost than its expansion. The second factor takes into account the normalized difference between estimated ($\bar{h}_k^e$) and predicted ($\bar{h}_k^p$) mean HTI of the same class-k traffic flows that the new request R belongs. Here, it is important to mention that we store the HTI of last few departed connections of every class in fixed size first-in first-out (FIFO) queues and calculate its mean value, which is termed as *estimated* mean HTI ($\bar{h}_k^e$). Note that traffic classes are defined by their bandwidth demands. Since the exact knowledge of HTI is not known for individual flows, the ML-based prediction is used to find in advance the *predicted* mean HTI ($\bar{h}_k^p$) of class-k traffic flows (connections). Thus, the difference factor ($|\bar{h}_k^e - \bar{h}_k^p|$) translates the accuracy of prediction into a weight in the cost function in Eq. (5.3). We explain a prediction mechanism in the next section.

5.3.3. Predicting Time-Varying Traffic with ML

Due to the properties and abilities like learning and self-X, where X is configuration, healing and optimization, AI is well suited for

the role of predicting traffic, classifications and making decisions. Among many AI techniques, statistical and learning methods have been widely used to predict time-varying traffic and to perform cognitive lightpath provisioning. Among the linear regression models, Auto-Regressive Integrated Moving Average (ARIMA) model and its variants were mostly used for time-series data prediction. However, non-linear regression models, especially learning methods such as support vector machine and neural networks, were evolved for predicting long time-series bursty data with non-linear arrival rate, which make them suitable for datacenter traffic prediction [49]. Since machine learning algorithms based on neural networks can extract informative features from raw data, many variants of it, such as deep/recurrent/convolutional neural network methods, have been utilized to predict (also classify) time-series data [75]. The prediction accuracy of these methods comes at the cost of complexity. Recently, [49, 93] applied back propagation neural networks to predict traffic in EONs to efficiently establish and mange lightpaths. However, back propagation and other neural network models could suffer from exploding and vanishing gradient problem – where the gradient of error functions decreases quickly without improving the learning process. Thus, Hochreiter and Schmidhuber [94] proposed a recurrent neural network architecture called long short-term memory (LSTM) to overcome these error back-flow problems. Due to capability of learning long-term dependencies, the LSTM network is very useful in classification, processing and prediction problems. Apart from an input and an output layer, a basic LSTM network has a hidden layer with one or more LSTM blocks, where each block has a memory cell for storing a value for long or short time periods, and gates (input, output and forget) for acting on signals they receive using activation functions. The time complexity of the LSTM learning model is linear $O(n)$ with samples n.

We use the LSTM network model to predict time varying traffic.

It should be noted that ML-based control and management would work efficiently under *periodic* or *known* circumstances or traffic patterns. However, under unexpected traffic patterns, prediction could be worse. To illustrate this fact, we first generated a time-varying mean and variance, and used lognormal distribution to generate a set of data, since the actual traces of DCs' traffic are not available in public domain. The time-varying mean values (data) can be obtained using the superposition of sinusoidal functions, as given by Eq. (5.4), where α is an amplitude constant, β_k and ϕ_k are frequency-dependent constants, and n is the number of frequency components (ω_k).

$$m(t) = \alpha + \sum_{k=1}^{n} \beta_k \times sin(\omega_k t + \phi_k) \tag{5.4}$$

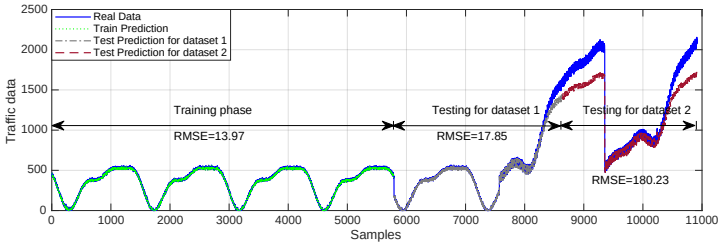

Figure 5.4.: Datasets used for training and testing, and their prediction. Only a fraction of data from different sets are shown here to highlight RMSEs [90].

We generated 43,200 traffic data samples per traffic class for training, validation and testing of the prediction model, out of which 67% is used for training, and remaining 33% for validation and testing of a LSTM network model, which consists of an input layer, a hidden layer with 4 LSTM blocks, and an output layer that makes a single value prediction. Each LSTM block has a memory cell for storing a

value for long or short time periods and gates (input, output, and forget) for acting on signals they receive using a Sigmoid and a *tanh* activation functions. We used an Adam optimizer to automatically set learning rate and to update network weights. To highlight the accuracy of prediction, Fig. 5.4 illustrates different RMSEs obtained during training and two sets of testing. Traffic profiles of "Real Data" are shown in blue color for all three phases (training, testing 1 and testing 2). As can be seen RMSE is very low for test dataset 1 and it is quite large for test dataset 2, since traffic profile used for testing dataset 1 follows the profile of real dataset used for training, but testing dataset 2 has different profile. Thus, it is essential to train ML-based models with different possible traffic profiles in order to obtain a good prediction, i.e., RMSE is close to zero.

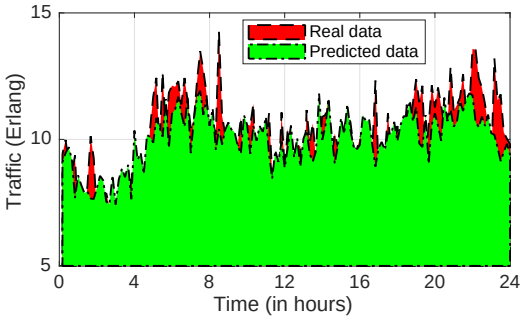

Figure 5.5.: Time-varying Real and predicted traffic [90].

Fig. 5.5 shows a more clear example of real and predicted data of time varying traffic during a day. We observe that although the predicted values differ from exact values, the LSTM model is able to predict the traffic profile. As described in the previous section, when exact knowledge of traffic is unavailable, the predicted mean HTI values for a time-interval can be used to calculate the MRLs of

traffic flows given their history and the costs of candidate spectrum paths and lightpaths for allocating resources to a new request.

5.3.4. Dynamic Resource (Re)allocation in ODCNs

In this section, we propose a dynamic ML-based resource provisioning scheme to route intra- or inter-DC flows and (re)allocate the resources like transponders, fiber cores, spectrum on-the-fly. This scheme, named as ML-based Resource (Re)allocation (in short ML-RAR), answers which lightpath or a spectrum path should be selected to provision a new class-k request? To this end, the ML-RAR scheme updates fixed size FIFO queues with the known HTI of departed flows to calculate the variable $\bar{h}_k^e$, i.e., estimated mean HTI of class-k traffic, and uses pre-computed predicted mean HTI for the current time interval, i.e., $\bar{h}_k^p$. Additionally, whenever a new request arrives, the MRL of each candidate lightpath is computed using the knowledge of lifetimes (history) of flows it serves, as described in Sec. 4.5. We describe the ML-RAR scheme in Algorithm 7, which is explained below.

First, κ-shortest (fiber-level) paths are computed for all origin-destination ($o-d$) pairs. Then, at any given time there are any of two possible events could occur: connection request arrival or departure. When a request arrives between a source s and destination d with demand b_k Gb/s, then the ML-RAR scheme tries to allocate resources, and admit it in the network. To achieve this, all possible candidate lightpaths and spectrum paths over allowed cores on its route are found and their costs are placed in sets $\mathbb{LP}$ and $\mathbb{SP}$, respectively. Note that Eqs. (5.3) and (4.5) are used to find costs for candidate lightpaths and spectrum paths, respectively. Among these two sets a minimum cost path, either an lp_i or sp_j, is selected for resource allocation of R i.e., $\mathbb{RA}(R)$. When a lightpath is selected for this purpose, then demand b_k is aggregated along other flows of lp_i, and if needed, a required number of additional subcar-

Algorithm 7 ML-based Resource (Re)allocation (ML-RAR) Scheme [90]

Given: Current time-slot Δ_t: network state (NS), guard band g, predicted traffic (λ and $\bar{h}_k^p, \forall k$), estimated mean HTI $\bar{h}_k^e, \forall k$, predicted blocking ϕ_B, and DF threshold γ_{th}.
if a connection request $R(o, d, b_k)$ arrives **then**
    Find all candidate LPs and SPs on allowed cores.
    Put costs of candidate LPs in $\mathbb{LP}$ and SPs in $\mathbb{SP}$.
    $\mathbb{RA}(R) \leftarrow min\{cost(lp_i), cost(sp_j)\}, \forall i \in \mathbb{LP}, j \in \mathbb{SP}$
    if $lp_i \in \mathbb{RA}(R)$ **then**
        Accept R and perform $\mathbb{RA}(R)$ along lp_i.
    else if $sp_j \in \mathbb{RA}(R)$ **then**
        Create a new lightpath over sp_j for $\mathbb{RA}(R)$.
    else if Reconfiguration is allowed **then**
        $\mathbb{RA}(R) \leftarrow \mathcal{DF}$–$\mathcal{AI}(R, NS, \phi_B, \gamma_{th}, g)$.
    end if
    If $\mathbb{RA}(R) = \emptyset$, block this request;
    Otherwise, setup a new lightpath for $\mathbb{RA}(R)$.
else // if a class-k connection (A_k) departs
    Free resources allocated to A_k.
    if class-k FIFO queue is full **then**
        Remove an HTI element from this queue.
    end if
    Add HTI of A_k to the class-k queue.
    Update $\bar{h}_k^e$.
end if

Algorithm 8 $\mathcal{DF}$–$\mathcal{AI}(R, NS, \phi_B, \gamma_{th}, g)$ [90]

if $\phi_B \geq \gamma_{th}$ **then** // C:= # of slices per core
    Try to Set free $\mathbf{b_k} = \lceil b_k/C \rceil + g$ consecutive slices on the route of R as follows. $\mathbb{RA}(R) \leftarrow \emptyset$.
    for $i = 1 : C - \mathbf{b_k} + 1$ **do**
        if $\mathbb{RA}(R) = \emptyset$ **then**
            Reallocate all lightpaths in spectrum interval $[i, i + \mathbf{b_k} - 1]$ on the route of R to other SPs.
        else
            return $\mathbb{RA}(R)$
        end if
    end for
    return $\mathbb{RA}(R)$
end if

riers is added to lp_i provided if the lightpath's bandwidth does not exceed the transponder's capacity. On the other hand, if a spectrum path sp_j is selected, then a new lighpath is provisioned over sp_j with bandwidth $\lceil b_k/C \rceil + g$ slices, i.e., including guard band g, where C is the capacity per slice. Therefore, a given number of guard bands g separates two adjacent lightpaths in the same fiber core to minimize inter-channel interference within a fiber-core, which is a major problem in EON.

However, it is not always possible to find either a lightpath or a spectrum path to admit a new request due to reasons like resource unavailability or spectrum fragmentation. Therefore, we utilize the same reactive (also known as on-demand) defragmentation (DF) method to reconfigure traffic flows, presented in the previous chapter. When a network operator allows reconfiguration to be performed, and if the request could not be accepted normally by methods in steps 3 to 9, then an intelligent reconfiguration process, described in Algorithm 8 as $\mathcal{DF}$–$\mathcal{AI}$ scheme, could be triggered (step 11). We call it intelligent because the DF process is triggered only when predicted blocking (ϕ_B) for the current time interval is equal or more than a threshold (γ_{th}). This avoids excess reconfiguration of lighpaths. For simplicity, the predicted blocking for time-slot Δ_t is calculated as $\phi_B(\Delta_t) = Tr_p(\Delta_t) \times BP(\Delta_{t-1})/Tr(\Delta_{t-1})$, where $Tr_p(\Delta_t)$ is predicted traffic in time-slot Δ_t, $BP(\Delta_{t-1})$ and $Tr(\Delta_{t-1})$ are the known blocking and traffic, respectively, in previous time-slot Δ_{t-1}. When the DF criteria is satisfied in step 1 of Algorithm 8, the reconfiguration process tries to set free $\lceil b_k/C \rceil + g$ number of slices to establish a new lightpath for R. However, even a reconfiguration process could not always guarantee the acceptance of a new flow request, and hence if resources can not be allocated to R, i.e, $\mathbb{RA}(R) = \emptyset$ even after the reconfiguration phase then it is blocked.

The novel feature of the ML-RAR scheme is that it not only takes prediction into cognizance but also extracts the information from

departed connections. When a class-k connection finishes its service, its resources are freed up. Interestingly, most of the requests do not advertise HTI at the time of their arrivals. However, when they depart, controller could calculate for how long they have been served, i.e., HTI becomes known for departed connections. Therefore, in steps 16 to 21 in Algorithm 7, fixed size (say 100) FIFO queues store the last 100 departed connections of every class, and updates estimated mean HTI $\bar{h}_k^e$, which is utilized in defining the costs of lightpaths in Eq. (5.3). The time and space complexity of the Algorithm 7 with reallocation is given in [90], can be obtained in a similar way as Algorithm 5, and it is non-polynomial in time.

5.3.5. Performance Evaluation

We evaluate the performance of the ML-RAR scheme with the time-varying traffic in terms of connection blocking, used transponders, and the number of reconfigurations being performed. We consider an intra- and an inter-ODCN topology [90] for the simulation. The simulation results are obtained using a discrete event simulation written in Java, and evaluated for 24 hours. Furthermore, to get the steady state simulation results, we simulate each traffic values for an hour, and start the next hour simulation with the previous network state (i.e., spectrum occupancy). Furthermore, we discard the results of an extra run hour at the beginning to get the network in a steady state. For the intra-ODCN, we consider only three hot top-of-rack (ToR) switches per aggregate optical switch, and each ToR switch is connected to several servers. For simplicity, the number of servers per ToR switch is not taken into account for the simulation. However, traffic is changed after every one hour such that enough requests are generated for every evaluation time-interval (1 hour) between ToR switches in intra-ODCN or between DCs in case of inter-ODCN topology. In inter-ODCN, each DC is directly connected with an optical switch, and requests are generated between any two

DCs. The ToR switches in intra-ODCN, and DCs in inter-ODCN topology are interconnected by multi-core fiber networks, where each link is bidirectional (consists of two unidirectional multi-core fibers) and each fiber consists of 3 cores, while each core has a total of 200 slices. We assume that a spectrum slice (mini-grid) occupies 12.5 GHz, and a spectrum-slice can support aggregated flows with a total demand of C =25 Gb/s using the QPSK modulation format. Additionally, we assume that the transponder's capacity is 10 slices, i.e., a lightpath can be established over a maximum of 10 slices. It should be noted that when a new lightpath is created to serve an incoming demand (b_k Gb/s), then the lightpath (logically) occupies $\lceil b_k/C \rceil + g$ consecutive slices, where g is the given number of guard bands (in slices) which generally help reduce inter-channel interference between two adjacent lightpaths within a fiber-core. However, we do not evaluate the effect of any kind of crosstalk within and among fiber-cores in this thesis.

In intra-ODCN simulation, we assume that each ToR-level Ethernet switch consists of 54 ports on the server side with three different line rates: 40 ports at 10 Gb/s, 10 ports at 40 Gb/s and 4 ports at 100 Gb/s. Thus, we consider three different classes of application requests with bandwidth demands b_1 = 10 Gb/s, b_2 =40 Gb/s and b_3 =100 Gb/s. As the configuration time in optical DCs is in the order of milliseconds, we focus on application requests with demand $b_k \geq 10$ Gb/s, which can be dynamically routed by setting up lightpath(s) upon their arrival and whose bandwidth requirements are comparable with line rates of Ethernet-ports of ToR switches. The same bandwidth classes and their arrival ratio (i.e., 40 : 10 : 4) are considered in the case of inter-ODCN simulation. We precompute first two shortest fiber-level paths between all ToR to ToR switches in intra-ODCN topology, and between DCs in inter-ODCN topology for routing of these three bit rate-based traffic flows (connections). Unless otherwise stated, we assume that all three classes

are latency-sensitive, thus they need to be routed immediately over pre-established lightpaths, if available or over spectrum paths by setting-up new lightpaths.

The connection arrival events are generated in each time-slot $\Delta_t = \{1, 2 \cdots, 24\}$ with exponential (parameter: λ per second) distribution, and an origin-destination ($o-d$) node pair (i.e., ToR–ToR in intra- and DC–DC in inter-ODCN) is selected uniformly. The overall request arrival rate (λ) is uniformly divided among traffic sources, i.e., among the number of ToR switches in the intra-ODCN, and among the DCs in the inter-ODCN. The arrival rate per class at any source node (i.e., λ_s^k) is proportional to the number of ports per class, which is in the ratio of $40 : 10 : 4$. Moreover, each connection departs after finishing its tasks, i.e., after its holding time, which is lognormally distributed with parameters: mean (real) HTI $\bar{h}_k(\Delta_t)$ and coefficient of variation (CoV) c_v. It should be noted that requests' arrival and departure events are generated at each source based on the real traffic per class (i.e., $f(\lambda_s^k, \bar{h}_k)$). To show the effect of HTI and its heavy-tailed characteristics, we fix mean arrival rate per source $\lambda_s = 10$ for inter-ODCN and $\lambda_s = 5$ for intra-ODCN topology for entire evaluation period (24 hours). The effect of time-varying arrival rate has also been evaluated at the end. The real and predicted mean HTI in the intra-ODCN case (see Fig. 5.6) is also scaled to half of that of inter-ODCN, such that blocking plots are visible and remain below the acceptable limit (10%) in most parts of the evaluation period for the ML-RAR scheme. Thus, the time-varying class-k traffic load is defined as $Tr_{type}^k(\Delta_t) = N \times \lambda_s^k(\Delta_t) \times \bar{h}_k^{type}(\Delta_t)$, where traffic type could be real or predicted based on whether mean HTI is real or predicted, and the number of traffic generator sources in the intra-ODCN and the inter- ODCN are the number of ToR switches ($N = 12$) and the number of DCs ($N = 5$), respectively. Furthermore, we calculate MRLs of connections using Eq. (5.2), which is a function of the already served time, i.e., the traffic his-

tory, its mean and CoV (or equivalently, μ and σ). It should be noted that for each class of traffic we maintain a separate FIFO queue to store HTI of last few (100) departed connections, in order to calculate estimated mean HTI $\bar{h}_k^e$. Unless otherwise stated, arrival rates are fixed as $\lambda_s = 5$ for intra- and $\lambda_s = 10$ for inter-ODCN topology, and connection requests are latency-sensitive and require immediate reservation, and connections are allowed to route over all fiber cores on their paths.

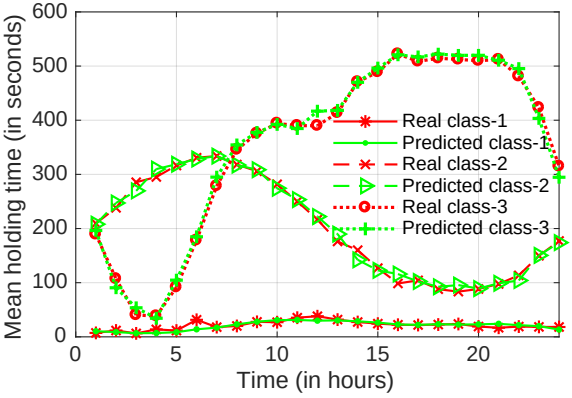

Figure 5.6.: Real and predicted mean HTI of three traffic classes [90].

To evaluate the performance of the ML-RAR scheme, we compare it with a first-fit (FF) spectrum allocation policy, which is simple yet very effective in a dynamic environment [37, 60, 72, 87]. However, it should be noted that the FF policy considered here as a benchmark first tries to find a first available lightpath that has sufficient free bandwidth to aggregate bandwidth demand of a new request, otherwise (i.e., when no lightpath is found) it tries to allocate a lowest indexed spectrum path among all cores to a new request. Additionally, the ML-RAR scheme has been evaluated for three different

scenarios: *known*, *predicted* and *unknown* traffic. When traffic is known, the exact (real) mean HTI, i.e., $\bar{h}_k(\Delta_t)$ is used in the cost functions in Eq. (4.5) (where Δ_t is not shown) to calculate MRLs of connections and lightpaths, and the predicted mean HTI $\bar{h}^p_k(\Delta_t)$ is same as $\bar{h}_k(\Delta_t)$ in Eq. (5.3). For predicted traffic scenario, $\bar{h}^p_k(\Delta_t)$ is used to calculate MRLs and the cost functions in Eq. (4.5) and Eq. (5.3), respectively. Interestingly, for unknown traffic case, neither exact nor predicted mean HTI is known, therefore, $\bar{h}^p_k(\Delta_t)$ term in Eq. (5.3) is ignored, i.e., $\bar{h}^p_k(\Delta_t) = 0$, and MRL of a connection is calculated as if HTI is exponentially distributed, i.e., MRL is independent of the passed time (memoryless property) and it is equal to the mean value of the unknown HTI. For the unknown mean HTI per class, we averaged the exact time-varying HTI values over 24 hours, and used this constant mean ($=\frac{1}{24}\sum_{\Delta_t=1}^{24}\bar{h}_k(\Delta_t)$) as MRL of every connection in all time-slots. Thus, the performance metrics (e.g., blocking) obtained for unknown traffic help us to show how important MRL is, especially in allocating resources to applications with heavy-tailed HTI distribution. Moreover, in all three traffic scenarios, the estimated mean HTI ($\bar{h}^e_k(\Delta_t)$) is calculated by extracting HTI from departed connections and used in Eq. (5.3) to calculate the cost of a candidate lightpath.

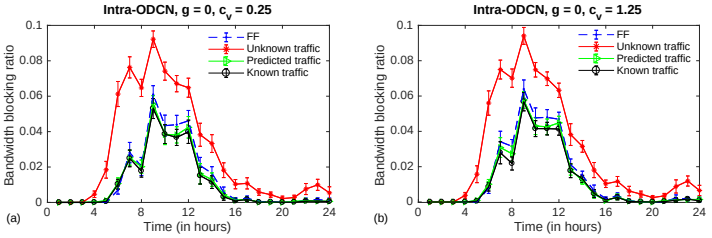

Figure 5.7.: BBR for unknown, predicted and known traffic in intra-ODCN topology are shown for g=0 slice and CoVs: a) $c_v = 0.25$ and b) $c_v = 1.25$ [90].

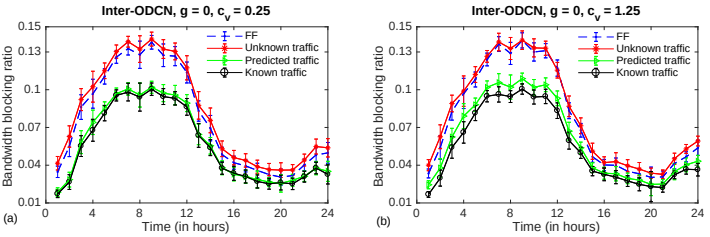

Figure 5.8.: BBR for unknown, predicted and known traffic in inter-ODCN topology are shown for g=0 slice and CoVs: a) $c_v = 0.25$ and b) $c_v = 1.25$.

We now show the BBR obtained for the ML-RAR scheme (without resource reallocation) under three different scenarios (*known*, *predicted* and *unknown*), and for the FF scheme in Fig. 5.7. The BBR is calculated per hour, and it is defined as the ratio of blocked bandwidth demands and total requested bandwidth demands of all arriving requests during an hour. The BBRs are shown with 95% confidence interval for the guard band ($g = 0$ slice) and CoVs ($c_v = \{0.25, 1.25\}$) for intra-ODCN in Fig. 5.7, and for inter-ODCN in Fig. 5.8. We make the following observations. i) Even in the absence of exact knowledge of traffic, the ML-based prediction could be efficiently utilized to allocate resources in ODCNs such that BBR is reduced, since the predicted traffic scenario outperforms the FF spectrum allocation policy and unknown traffic, and it is quite close to the BBR of the known traffic case. ii) The utilization of traffic history or active lifetimes of connections and the predicted or exact knowledge of traffic are absolutely necessary for an efficient resource allocation scheme, as we can see in Figs. 5.7 and 5.8 that BBR is quite high in case of unknown traffic. iii) The MRL effect alone can be seen for higher CoV ($c_v = 1.25$) as compare to $c_v = 0.25$. We observe that increasing the CoV slightly increases BBRs in all traffic scenarios and in both schemes, since more connections with larger

holding times are generated even for smaller mean HTI if CoV is large. However, the increment in BBR is more visible in cases other than the known traffic scenario, that either uses prediction (in predicted traffic case) or simply does not take active lifetime of connections (history) into account (in unknown traffic and the FF scheme), which results in inaccurate estimation of MRLs. Generally, the increasing the number of guard bands $g \geqslant 1$ increases the BBR, as we show in [90]. Furthermore, the number transponders per traffic source used for creating new lightpaths is shown in 5.9. Notice that

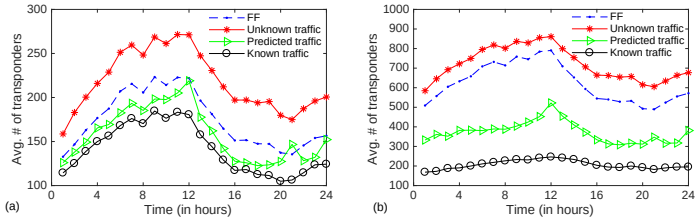

Figure 5.9.: Average number of transponders per source in a) intra-ODCN and b) inter-ODCN topology are shown for g=0 slice, $c_v = 1.25$ [90].

the unknown traffic uses relatively more transponders, and inter-ODCN topology uses much higher number of average transponders per hour than the intra-ODCN, since the offered traffic in the former is four times, and also due to the fact that the inter-ODCN topology provides more disjoint routes than the intra-ODCN topology, which also helps in limiting blocking in the inter-ODCN despite serving much higher traffic.

It is evident from the previous discussion that traffic prediction is an important factor in handling resources intelligently and efficiently in both types of ODCN networks or in any networks in general. However, as shown in the previous chapter and in [60], resource reallo-

cation could further improve the BBR performance. Furthermore, the reactive (on-demand) defragmentation (DF) could be triggered intelligently with the help of predicted BBR for future time-slot. Fig. 5.10 depicts the BBR obtained in the ML-RAR scheme for predicted traffic scenario with and without enabling the DF-AI feature (explained in Algorithm 8). Furthermore, we introduce a predictive but static DF-AI triggering condition, i.e., whenever the predicted BBR (ϕ_B) crosses the pre-defined threshold (i.e., $\phi_B \geq \gamma_{th}$), then the DF-AI is triggered. Fig. 5.10 shows BBRs for different threshold in intra- and inter-ODCN topology, i.e., $\gamma_{th} = \{0, 0.01, 0.05\}$. It should be noted that $\gamma_{th} = 0$ represents a complex and opportunistic case: whenever a connection could not be accepted without the reallocation phase (DF-AI), the DF is triggered. Thus, it represents an unintelligent triggering solution leading to the lower bound on the BBR obtained under the ML-RAR scheme with or without the DF phase. Also, we restrict the reconfiguration of lightpaths within the same fiber-core where they are provisioned to reduce the complexity involved in a DF operation and also to limiting the traffic disruption caused by reconfigurations of lightpaths over different spectrum, fiber-cores or paths during a DF operation. It can be seen that enabling the DF-AI scheme could reduce blocking by almost an order of magnitude for some time-intervals ($\Delta_t = \{6, \cdots, 12\}$), especially in intra-ODCN (likely due to its topology and lesser number of disjoint routes which cause more spectrum fragmentation). Moreover, the BBR for $\gamma_{th} = 0.01$ in intra-ODCN and $\gamma_{th} = 0.05$ in inter-ODCN oscillate between the BBR of predicted traffic without (w/o) the DF case and the DF-AI case with $\gamma_{th} = 0$. At the same time, it should be noted that a static DF triggering threshold $\gamma_{th} > 0$ and predicted BBR (ϕ_B) have some flaws, which is reflected on BBRs during time-slots $\Delta_t = \{6, 8, 14\}$ wherein the DF-AI is not triggered for $\gamma_{th} = 0.01$ (see Fig. 5.10a) despite having moderate load in the intra-ODCN network. The oscillation (a sharp

increase and decrease) problem can be overcome if BBR prediction and blocking threshold are adaptive in nature and consider the following suggestions. First, while predicting the BBR it must be taken into account that whether BBR obtained for the previous interval is obtained with or without DF-AI scheme. Secondly, if the BBR obtained in the previous interval is with DF-AI, then the threshold must be slightly lowered for the current interval. Thus, we recommend an intelligent dynamic threshold rather than a static threshold for triggering the DF-AI process. The similar effect is visible in the average number of reconfigured lightpaths per hour in successful DF operations in Fig. 5.11. Here, a successful DF operation represents a DF event that could reallocate some lightpaths to accept a new request. Furthermore, a trade-off between blocking performance and average reconfigured lightpaths can be seen for both topology. Thus, it is up to network operators to utilize the traffic and corresponding blocking prediction to obtain a balance between blocking and reconfigurations, and eventually activate or deactivate some resources (e.g., transponders) in advance.

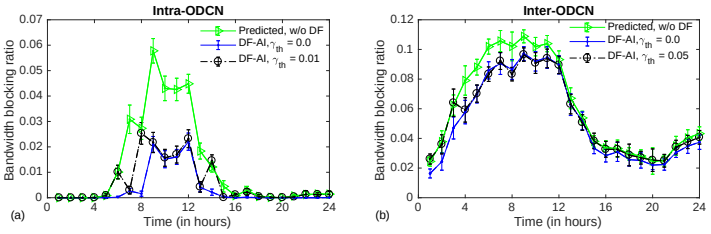

Figure 5.10.: BBR for predicted traffic and ML-based defragmentation in a) intra-ODCN and b) inter-ODCN topology are shown for g=0 slice, $c_v = 1.25$ [90].

Finally, to highlight the importance of prediction accuracy, we calculate the predicted mean HTI by adding absolute and percentage

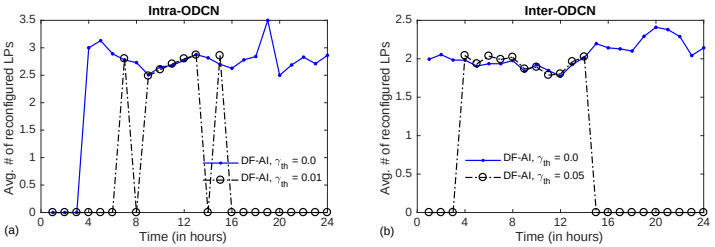

Figure 5.11.: Number of reconfigured LPs per DF per hour for predicted traffic in a) intra-ODCN and b) inter-ODCN are shown for g=0 slice, $c_v = 1.25$ [90].

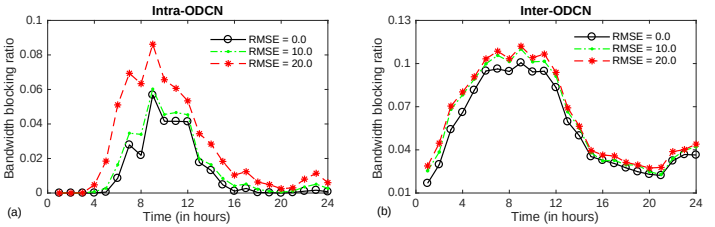

Figure 5.12.: BBR in a) intra-ODCN and b) inter-ODCN topology are shown for g=0 slice, $c_v = 1.25$, for predicted traffic with fixed arrival rates and absolute increase in RMSE [90].

increase in RMSE to the exact mean HTI. Again, remember that the real traffic is used to generate connections during simulation, and the predicted mean HTI is used for spectrum and lightpath cost assignment in Eqs. (4.5) and (5.3). The exact (real) time-varying mean HTI for intra-ODCN is shown in Fig. 5.6, and it is doubled for inter-ODCN topology. In Figs. 5.12 and 5.13 we present BBR with absolute and percentage change in RSMEs, respectively for intra- and inter-ODCN topology. Fig. 5.13 is obtained by for a time-

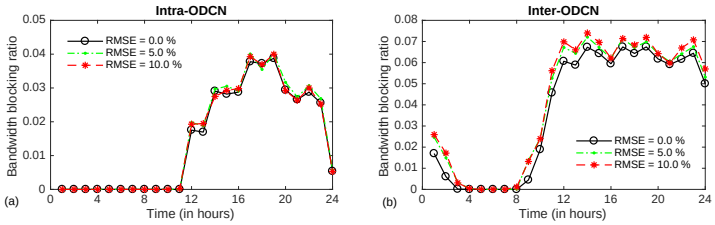

Figure 5.13.: BBR in a) intra-ODCN and b) inter-ODCN topology are shown for g=0 slice, $c_v = 1.25$, for predicted traffic with time-varying arrival rates [90].

varying arrival rate. For intra-ODCN, the arrival rate per source is varied during the evaluation period as $5-3\sin(2\pi(\Delta_t-1)/24)$, where time-interval $\Delta_t = \{1, 2, \cdots, 24\}$, and the arrival rate per traffic source is doubled for the inter-ODCN topology. It is interesting to see that the performance of the ML-RAR scheme depends on how well ML predicts the traffic, i.e., BBR increases with RMSE. Moreover, when the error is in the order of mean HTI, i.e., with 100% RMSE, which is the case for class-1 (mean HTI $\sim$ 10 seconds) in Fig. 5.12, then the BBR could be worse. In contrast, when the error is relatively low (in Fig. 5.13), then the effect of RMSE is not noticeable. Thus, in an unpredictable scenario when ML would not able to closely predict future event characteristics, ML-based prediction might lead to inefficient management of optical resources.

5.4. Resource Allocation in Fiber-Wireless Networks

As a second use case of ML techniques, we propose and study an early warning system using the Internet-of-Things (IoT)-fog-to-cloud system comprising of IoT devices (cameras, sensors), Base Stations (BSs) and the cloud. In the proposed system, we first detect movements of animals at sensor nodes, then transfer sensor data

to fog nodes (at BSs) over wireless sensor aggregation networks, and finally to the cloud for further processing over a passive optical network. The information processed is then distributed to humans (or vehicles) that are living (or passing by) near areas of animal habitats. Since an early warning system is a latency-critical application, the main question we try to answer is whether to transfer all data to higher layers for better inference (with respect to classification or alert accuracy) or to process data at edge devices before transferring to the fog nodes (e.g., base stations). At the same time, however, processing of data at its origin, i.e., edge devices (that would result in transferring lesser data) could result in false alarms being generated due to the limited store and computing power of edge devices. Thus, first we try to find the trade-off between communication cost and computation/processing cost through an experimental setup by showing the average amount of current drawn (or energy consumed) during image capturing, processing and transmission. Then, we show the trade-off between end-to-end latency (time between animal appearance to notification) and data volume transferred for each animal detection event in our distributed computing and AI-enabled early warning system through a simulation. Our experimental results show that data processing at end devices could save around 57% energy.

5.4.1. Fiber-Wireless Network

The dependency on cloud-based services is growing, and so is the amount of network traffic from IoT devices to the cloud, and vice versa. Thus, hybrid fiber-wireless networks present a viable solution to ensure high capacity, high flexibility and low-cost broadband access to cloud-based services [95]. A typical architecture of a Fiber-Wireless (FiWi) network is an integrated passive optical network and wireless networks, where each optical network unit is connected to several wireless nodes (e.g., sensors, smartphones) through a wireless

gateway (e.g., base station).

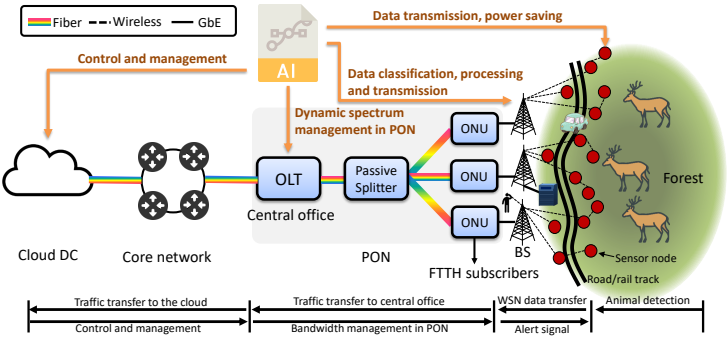

Figure 5.14.: An architecture of FiWi network to carry cellular, FTTx, and animal traffic; and applications of AI techniques in different network subsystems are shown [96].

Fig. 5.14 illustrates an envisioned FiWi architecture that integrates a Passive Optical Network (PON) and Wireless Sensor Network (WSN), and shows an example of network subsystems in all three layers: edge (sensors), fog (base stations, optical network units), and the cloud. As a main block of a FiWi network, a PON consists of an Optical Line Terminal (OLT) at the service provider's central office, a passive splitter and Optical Network Units (ONUs). An OLT is a terminal equipment that includes a gateway router to connect PON to the fiber backbone or core networks. Additionally, its main function is to define transmission window and allocate bandwidth to ONUs which are near the end users. ONUs are expected to serve heterogeneous traffic of end-devices, since they would connect to wired, e.g., fiber-to-the-home (FTTH) subscribers, and wireless traffic through BSs in the 5th generation mobile networks. A single or a group of passive splitters allows to share a PON network among many end users by means of splitting power in Ethernet PON and/or

demultiplexing wavelengths in WDM-PONs.

We use an OFDM-PON architecture, where each ONU processes only its assigned fraction of the OFDM spectrum generated by the OLT. This is because of its known better performance in terms of latency and synchronization compared to other PON technologies [97]. Also, dynamic bandwidth allocation techniques used in the OFDM-PON fully exploit the spectrum flexibility. This motivates us to use the system such that we allocate subcarriers' bandwidth based on the traffic volume of each class (cellular, FTTH and animal) within each ONU. At the same time, however, segregation of bandwidth requires the classification and prediction of traffic volumes, which in fact can be done through AI techniques. Various AI-based applications can be envisioned in Fig. 5.14, including the management of network devices and bandwidth allocation. For instance, instead of running a centralized bandwidth management of PON subsystems at the cloud, a central office (OLT) could dynamically allocate spectrum in PON subsystems based on the traffic predicted at its ONUs. This would help improve the bandwidth management and also latency.

In Fig. 5.14, wireless sensor nodes (edge devices) form an aggregation network along the road (e.g., to mimic a point of human-animal conflict) and they are wirelessly connected with the so-called sinks, denoted as base stations (BSs). Whenever a WSN node detects animals' presence, it tries to send sensed data (e.g., picture/video) of its surroundings to a nearest BS. The energy aspect is also relevant for systems that are not connected with power grid and are, for example, either battery-based, or solar-powered. Another advantage of AI techniques is that they could also relieve the network congestion by avoiding the unnecessary data transfers, which also improves the end-to-end delay of those transfers that need to go over the network. On the other hand, a limited processing and storage resource pool at sensors would also have negative impact on the accuracy

of classification and prediction. In contrast, a cloud-based infrastructure connected to the ONUs could deduce information from the animals' raw data with more accuracy in classification and prediction. This makes it interesting to investigate the role and placement of AI-based methods in animal welfare and human-animal cohabitation systems based on FiWi.

5.4.2. The Role of Machine Learning

Machine learning helps particularly in identification of animals' movement using image classification algorithms, and predicting animals' movement or traffic in general, correlate it with human movement and traffic, or classifying animals' behavior. Additionally, based on the network data traffic predicted, a hybrid approach of processing and communication could be used. In other words, animals' data could be processed at the source sensor (or end device) when the animal movement is predicted as more likely to happen. On the other hand, when animal movement is less likely to happen, during that period, most of the data collected could be transferred to fog nodes (e.g., server, mini-cloud) for more accurate processing, since a fog node with enough compute and storage capability can run heavyweight deep neural network models (e.g., ImageNet [98]) that generally have more hidden layers and neurons than the lightweight models. At the same time, and as previously mentioned, attention needs to be paid to the accuracy of the classification and the prediction, which is a challenge.

In the OFDM-PON architecture, machine learning algorithms can be used to split optical resources flexibly based on predicted traffic demands of heterogeneous services. For example, in a PON, upstream flow scheduling at ONUs is more critical due to lower data rate (e.g., 2.5 Gb/s) as compared to downstream traffic at OLT with higher data rate (e.g., 10 Gb/s). Moreover, an ONU needs to allocate OFDM subcarriers in an OFDM-PON or time slot in an Ethernet-

PON to various types of application demands based on their requirements (latency, bandwidth). As observed in the literature, peak and low of various traffic patterns could occur at different times during a day. Thus, learning-based traffic prediction could allocate just-enough bandwidth to data traffic demands [90]. In 5G, for example, cellular connections would need to satisfy latency in the order of milliseconds. Therefore, a set of dedicated but flexible frequency bands and slots must be assigned for latency-sensitive applications. More importantly, a lightweight image classification model, such as MobileNet [99], can be run on power and resource constrained end devices, and they could be helpful in identifying animals' presence or absence, and other related information.

To address the issue of the said granularity of traffic patterns, we consider multi-class (three) traffic scenarios and use machine-learning-based traffic prediction to allocate bandwidth as per traffic demands of each class. Specifically, we use the LSTM model for the time-series traffic prediction, which is shown to be useful in predicting time-varying traffic in Section 5.3. Additionally, for animal identification on end devices (Raspberry Pi) connected with sensors, we use a MobileNet model [99] which is proposed recently for image processing on resource and power constrained devices, such as smart phones. The MobileNet model utilizes Convolutional Neural Network (CNN), wherein depth-wise separable convolutions are used to build lightweight deep neural networks.

5.4.3. Computation vs. Communication

With the increasing availability of low cost and computing capable end devices, there is a paradigm shift from cloud computing to edge or fog computing today. Nevertheless, the end devices are mostly resource constrained and also power constrained when they are remotely located. Therefore, an important aspect in an animal-human cohabitation system based on an IoT-fog-to-cloud architecture is to

analyze the trade-off between the computation cost and the communication cost in terms of energy consumption at end devices. In this section, we show this trade-off through an experimental setup and measurement results. Throughout this section, we assume that an edge (end) device, e.g., Raspberry Pi (R-Pi), has some computational capability and it can also communicate to other edge devices or upper layer nodes, e.g., base stations or servers. The end devices are integrated with various sensors.

Experimental Setup and Measurement Results

The experimental setup is shown in Fig. 5.15. A R-Pi 3 (model B) as an end device is connected to a passive infrared (PIR) sensor to detect the movement of animals, and the PIR sensor is used to wake up a 5 megapixel (MP) camera module to capture images of animals. Additionally, we utilize a low-power ZigBee communication protocol to transfer data between an end device and a server (PC) without any packet retransmission. For which, we use two XBee transceiver modules, one is configured as an end device and connected to the R-Pi, and another is configured as a coordinator and it is connected to the server situated a few meters apart. Since the R-Pi can process images with the help of a CNN image classification algorithm, we evaluate two scenarios: (i) *capture-transfer*, and (ii) *capture-process-transfer*. The capture-transfer scenario captures an image at the end device and then transfers its binary data to the server. On the other hand, the latter scenario captures an image, processes it to detect whether animals are present or not and their types, and finally the end device transfers a message about the animals' presence in its vicinity. For the image processing and classification on the R-Pi, we install a pre-trained CNN model (trained on a PC) using a TensorFlow framework [100] which is widely used in the community on the R-Pi. The experimental setup to measure the current drawn during the operation of the two scenarios is shown in Fig. 5.15. We

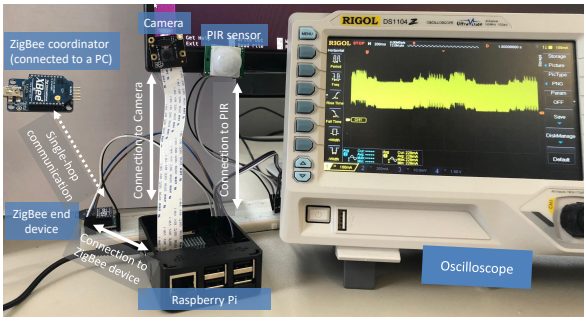

Figure 5.15.: An experimental setup of a single-hop communication between two ZigBee (XBee) transceivers, one configured as an end device (WSN node) and another as a coordinator (sink). A Raspberry pi (R-Pi) as an end device is integrated with a camera, a PIR sensor, and a XBee transceiver [96].

use an oscilloscope to measure the amount of current drawn by the R-Pi module. The current is measured from the voltage drop across a very low shunt resistor (1 Ohm), which is connected in series to the R-Pi module.

To find the trade-off between the communication and the computation costs in terms of average consumption of energy, let us evaluate individual parts associated with both scenarios. Fig. 5.16(a) shows the average current required to capture an 89 Kilobytes (KB) image, which is triggered when a PIR sensor detects a movement around it. The amount of current drawn depends on various parameters: camera resolution, picture size, color quantization, image compression (raw, jpeg, png), etc. For this measurement, we used a 5 MP camera and set it up to capture 299 $\times$ 299 pixel color images. We used jpeg as a compression format. The average time required for the camera to capture an 89 KB image is around 0.84

s. The second part in the capture-transfer scenario is the transmission of the image captured in the previous step to the server. The energy consumed (or current drawn) by the system during the transmission varies depending on the amount of data transmitted, distance between the sender and the receiver, and the selected working mode, since the XBee transceiver module can work in different modes: *coordinator, router* or *end device*. In this setup we evaluate a single-hop transmission, i.e., one XBee is configured as an end device (integrated with the R-Pi) and another as a coordinator (attached to a PC by a USB connector). Fig. 5.16(b) shows the average current drawn by the R-Pi module while transmitting an 89 KB image. The transmission time for this operation is around 19.33 s. Therefore, the average transmission rate achieved in our experiment is only ∼37 Kbit/s, which is much lower than the maximum data rate that a ZigBee (or IEEE 802.15.4) end device can transfer, i.e., 250 Kbit/s in a 2.4 GHz spectrum band. Moreover, the average per packet latency is around 22 milliseconds if we ignore the control packets, which is negligible as compare to 89 KB data, since each data packet carries approximately 100 bytes of payload, and the transmission time of 89 KB data is around 19.33 s. Therefore, the energy consumed in the capture-transfer scenario at 5 V is approximately given as $5 \times (0.44 \times 0.84 + 0.39 \times 19.33)$ =39.5 J, which uses the average current 0.44 A for the image capturing, and 0.39 A for its transmission.

For the capture-process-transfer scenario we process a captured image before transmitting some relevant information. Fig. 5.16(c) shows the average current required to process an image. For the image classification (processing), we used one of the widely used CNN models on constrained devices: MobileNet. MobileNet is a small size CNN that can support different input image resolutions. The classification accuracy improves with higher resolutions, however, at the expense of more processing time. The model is retrained to recog-

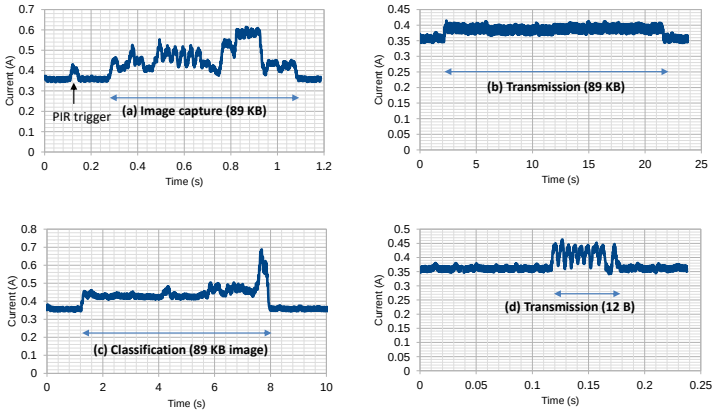

Figure 5.16.: Average drawn current in: (**a**) capturing an 89 Killo bytes (KB) image; (**b**) transmission of 89 KB image; (**c**) classification of the same captured image (89 KB); (**d**) transmission of processed data of size 12 bytes [96].

nize two classes of images: the presence and the absence of animals inside the captured image. The classification time for an 89 KB image is around 6.8 s. Finally, a 12 bytes data message about the presence or absence of animals and their types are transferred from the end device to the server, for which the average drawn current is shown in Fig. 5.16(d), which takes 0.05 s. The total runtime for the three operations shown in Figs. 5.16(a),(c),(d) is around 7.6 s, and the energy consumed in the capture-process-transfer scenario at 5 V is around $5 \times (0.44 \times 0.84 + 0.44 \times 6.8 + 0.40 \times 0.05) = 16.9$ J, which results in ∼57.2% energy saving as compare to the capture-transfer scenario. Thus, we can conclude that the communication cost in power constrained devices could be very high if they require to transfer more data. Notice also that the necessary time to realize the capture-transfer scenario is much higher (around 20 s)

than the second scenario (7.7 s). Therefore, it is better to compute and process at the end devices, if possible, and transfer only relevant information to the server. It should be noted that the energy consumption can be further reduced by processing data near sensor nodes using lightweight machine learning algorithms, such as hidden Markov model, support vector machine, etc. However, the classification accuracy would be reduced, which might result in generating more false alarms [101].

5.4.4. A Framework for an Early Warning System

To analyze the performance of the proposed network system, we design the following subsystems, earlier illustrated in Fig. 5.14: (i) animal detection; (ii) message transfer to fog nodes (BSs); (iii) alert vehicles/humans, when necessary; and (iv) bandwidth allocation subsystem in the OFDM-PON, located between fog nodes and the cloud. Let us now present and discuss each subsystem one-by-one.

Animal detection subsystem is the most crucial part of the animal-human cohabitation system. As energy is a major issue in remote sensors, only a few sensors need to continuously detect the animal crossing events by sensing surrounding movement, pressure, vibration etc. At the same time, disturbances caused by the wind and other events that can create false alarms (such as large birds) need to be differentiated from relevant animal events. It has been shown earlier that fiber- or infrared-based sensors are most suitable for this purpose [102]. Once animals are detected, the animal triggering sensors (e.g., PIR) would wake up other sensors (camera) to capture mainly images, which then needs to be processed to find animals' size, position, speed, numbers, direction and other specific features. As we discussed before, the processing of raw data either at origin (end devices) or at higher layers (fog or cloud) depends on the capability of sensor nodes as well as on network ability to transfer data without congestion. Even though a capable end-device with various

sensors could process the raw data, in case of images, the processed data need to be sent to fog nodes, which can reprocess it to decide whether to generate an alert message or not.

The second subsystem is responsible for transferring of the processed or unprocessed data to fog (BS) nodes. For this purpose, whenever a sensor detects movements of animals, it tries to send raw data (e.g., pictures, GPS signal) or just relevant (processed) data to a sink node, i.e., BS. Whether to send raw or only relevant data depends on the AI running capability of end devices. Let us describe various sub-tasks of the first two subsystems. It starts with animal detection, then images are captured, and data is processed at end devices if they could run processing algorithms. At last (un)processed data is transferred from end devices to BSs. We utilize a low-power ZigBee communication protocol to transfer data from sensor nodes to BSs, which specifies a maximum 250 Kbit/s data rate and transmits 128 bytes per packet (where approximately 100 bytes can be the data payload) over unlicensed spectrum of 2.4 GHz. As the sensor nodes are assumed to be deployed along rail/road tracks, and ZigBee utilizes the shortest path to send a packet from a sensor to a BS, we use tree topology in Riverbed (earlier known as OPNET) Modeler simulator [103]. The simulator models three different kind of nodes: *End Devices*, *Routers* and *Coordinators*. End Devices only act as sensor nodes without routing capabilities, while routers can play both sensor and forwarding roles. Finally, coordinators, acting as sinks, are responsible for the connection among all nodes, and are also capable to act as sensor or router nodes.

The third subsystem is dedicated to transfer alert messages to humans/vehicles, and to this end, it may need to be designed to process the data further at fog nodes (BSs) and to trigger the alarm (notification) signal, when necessary. Although the alert signal is generated at BSs, animals' related traffic still needs to be sent to the cloud for long-term statistics and efficient execution of notifica-

tion messages by sharing them with neighboring BSs. Therefore, we assume that the fog nodes (BSs, ONUs) locally connect to servers or a mini-cloud for processing and classification of various traffic (cellular, FTTH, animal). In this way, the mini-cloud is also the part of a fog computing system, able to replacing the cloud functions in the system, which is fog's salient feature.

As can be seen in Fig. 5.14, each BS is connected directly to an ONU through a Gigabit Ethernet interface, and the OLT schedules the ONU's traffic by allocating required bandwidth as per their overall traffic demand. However, it should be noticed that various types of traffic share the ONU's upstream bandwidth. Thus, as mentioned in Section 5.3.3, the forth subsystem utilizes the predicted traffic to dynamically allocate bandwidth of OFDM subcarriers per traffic class at ONUs. A new service request is routed over an existing lightpath that has free sufficient bandwidth, otherwise a new lightpath is created to carry traffic of a new request if sufficient resources (spectrum, transceiver) are free. It should also be noted that the cellular and other connection requests are blocked if resources are not sufficient to satisfy the demand. The exception to this is animals' traffic, as it needs to be transferred to the cloud for long-term statistics. In our approach, the animal's traffic is transferred either immediately or whenever resources are free, therefore we assume that a queue can be maintained at ONUs/BSs with sufficiently large size to store animals' data.

5.4.5. Performance Evaluation

We evaluate the performance of an integrated IoT-fog-to-cloud early warning system by three performance metrics: average end-to-end per packet latency, notification time, and connection blocking. The average end-to-end packet latency is measured as the average time required to transmit a sensor data packet from an edge device to a BS, which mainly depends on the congestion in WSN networks. The

notification time is measured as the average time difference between the animals' detection and the alert message disbursed to the passing vehicles/humans. The connection blocking happens in the OFDM-PON when ONUs tries to reserve required bandwidth for a certain duration to transfer data of a traffic class.

The scenario we analyze using the Riverbed Modeler simulator consists of three end-devices and two routers per coordinator (BS), as shown in Fig. 5.17. End-devices act as sensor nodes, and routers act as relay as well as sensors, detecting the animal presence and capturing images of surroundings to deduce animal information, including animal position, movement direction, speed etc. Data is split into packets of 100 bytes payload (ZigBee), which are queued for transmission. The packets are consecutively forwarded to the coordinator acting as a sink (BS). The OFDM-PON as an optical access network is simulated with 16 ONUs. We assume that a WSN network with 5 sensors is connected to each base station with shortest routing path, and each base station is directly connected to an ONU. The WSN topology is an aggregation-based network, as shown earlier in Fig. 5.14, where the BSs act as sink nodes. Wireless link capacity is assumed to be 250 kilobits per second (Kbit/s) in ZigBee protocol without any packet retransmission. We assume fiber capacity as 4 THz in the OFDM-PON, which is divided among 320 spectrum slices, each with granularity of 12.5 GHz. Each spectrum slice could support a maximum of 25 Gb/s using a QPSK modulation format. All 16 ONUs share equal proportion of fiber upstream bandwidth, i.e., each ONU can reserve 20 spectrum slices.

We evaluate the performance of our animal-human cohabitation system by means of latency observed (i) with AI, that transfers the processed data from sensors to BSs (ii) without AI, that transfers raw sensed data, and (iii) the so-called *hybrid approach* that transfers most of the data during low traffic condition and only relevant information during busy hour.

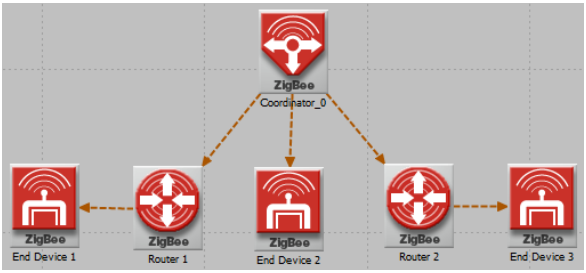

Figure 5.17.: WSN topology per BS simulated by Riverbed Simulator [96].

We consider three classes of services: cellular or call (class-1), FTTH (class-2) and animal traffic (class-3). Fig. 5.18 illustrates the real and the predicted time-varying mean traffic arrival rates of these three classes. The class-1, class-2, and class-3 connection requests at BS or ONUs require 1 Gb/s, 10 Gb/s, and 250 Kb/s, respectively. It should be noted that mean traffic arrival rates vary with time t, and the traffic per class is calculated as $Tr_k(t) = (N * \lambda_k(t)/\mu_k)$, where N is the number of class-k traffic sources, λ_k is the class-k mean arrival rate at the source of traffic, and $1/\mu_k$ is the mean holding (service) time of class-k connections that reserves upstream bandwidth between ONUs and a central office. We assume traffic arrival for each class as a Poisson process with time-varying mean arrival rates shown in Fig. 5.18, where the mean arrival rate of a class-k traffic is changed every hour. The mean holding times of class-1 and class-2 traffic are assumed to be exponentially distributed with mean 10 minutes. For each animal appearance event at a sensor, its corresponding ONU tries to reserve a 250 Kbit/s bandwidth between the ONU and a central office. Therefore, the mean holding time required to transfer packets of each animal event at ONUs is kept as a constant (40 milliseconds, i.e., 0.00067 minute), since the data

volume (processed) of each animal connection that needs to be transferred from an ONU to the central office is fixed as 1 KB, and the required bit rate is assumed to be 250 Kbit/s. The animal appearance events are generated at sensors (end devices), and the cellular and the FTTH arrival events are generated at each BS/ONU. The number of simulated arrival events for every hour varies with the time, since the arrival rate for each class changes after every hour. Furthermore, we are focused on upstream reservation, therefore, all results presented here are based on upstream reservation from edge devices to BSs to central office. We maintain a sufficiently large fixed size queue to store data packets of animal class at each ONU. Other latency-sensitive classes of traffic (cellular and FTTH) are either served immediately or blocked.

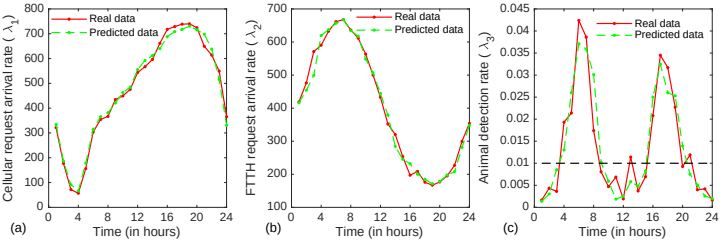

Figure 5.18.: Real traffic vs. predicted traffic of three classes of services, namely cellular, FTTH and animal are shown in (**a**,**b**,**c**), respectively [96].

Fig. 5.19 illustrates the average per packet delay (latency) between sensors and BSs of animal traffic under three data transmission scenarios: 1 KB, 100 KB and hybrid (i.e., a mix of 1 and 100 KB) using the ZigBee protocol. These scenarios also depict the case of the data processing at different layers: edge (sensors), fog (BS) and a hybrid approach, i.e., processing either at the edge or at BSs. We could also compare these scenarios with the cloud processing, where

raw sensed data is sent to the cloud for the processing and decision making. However, we ignore this scenario, as we did not simulate the core networks and the cloud. However, it is obvious that the latency obtained in the cloud processing will be much higher than that of the other three scenarios presented here. Furthermore, we assume that the given specified data volume is transmitted for every animal detection event at sensors. In the first scenario, we assume that sensors are attached to a processing unit (e.g., R-Pi), thus they run a simple machine learning model (e.g., MobileNet) to detect and classify animals' related data and transmit only 1 KB of data (around 10 data packets) for every detection, as we have also shown through an experimental setup in Section 5.4.3. In the second scenario, end devices (sensors) leave the processing task to the fog layer (BSs) and transmit 100 KB data (around 1000 data packets) to BSs. The hybrid case, on the other hand, takes predicted animal traffic into the account to decide whether to process data at the edge layer (sensors) or at the fog layer (BSs). The decision threshold is shown by a black dotted line in Fig. 5.18c, and it could be set or even varied by monitoring the congestion or by predicted animal traffic. In the hybrid case, we enforce processing at sensor nodes (1 KB transmission) when the predicted arrival rate is higher than the threshold, otherwise the end devices transfer 100 KB data per animal detection.

From Fig. 5.19 we observe that when end devices process the raw data and transfer only 1 KB data to BSs per animal detection event, then the average packet latency varies between 26 to 30 milliseconds. Thus, in this case it can be deduced that WSNs are not congested (i.e., queuing delay is negligible). Therefore, we can compare it with the one-hop transmission in experimental setup in Section 5.4.3, where the average per packet latency is obtained as 22 milliseconds. We observe that both results are comparable. However, it also depends on the distance between a transmitter and a receiver. Unlike the experimental setup, the maximum distance between end devices

and routers/coordinator is kept in the simulator environment, so a small propagation delay also contributes to the latency obtained in the simulation environment. Furthermore, the effect of congestion on packet latency can be seen in the 100 KB transmission case, and the latency increases by two order of magnitude than that of the prior scenario (1 KB transmission). Interestingly, the average packet latency in the hybrid scenario can be reduced from a few seconds to milliseconds during the congestion hours of the day by applying an intelligent control mechanism using the traffic prediction.

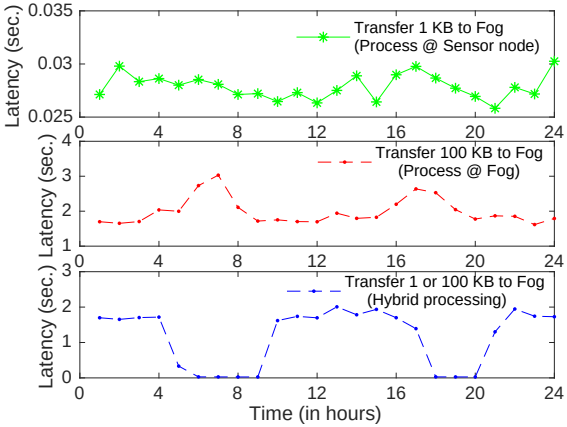

Figure 5.19.: Average end-to-end packet delay from sensors to BSs [96].

Fig. 5.20 shows the approximate average notification time, which is the average value of the time difference between the arrival of an animal detection event and an alert message disbursed to the passing vehicles/humans. The notification time is calculated as the sum of the data transmission time, the average packet latency and the processing time. We ignore the latency in disbursing alert messages

from BSs to passing vehicles, since it is negligible as compared to other latencies and remains same in all three scenarios compared here. Furthermore, we assume that processing times at sensors and BSs are 5 s and 1 s, respectively. ZigBee transmits 100 bytes data per packet at a maximum rate of 250 Kbit/s, thus the total transmission times of 1KB (10 packets) and 100 KB (1000 packets) data are approximately 0.04 s and 4 s, respectively. As can be seen in Fig. 5.20, the notification time in the second scenario (i.e., processing at fog and 100 KB data transmission) varies from 6.5 s to 8 s as compare to the first scenario (processing at sensors and transferring just relevant information 1 KB), which is around 5 s. It should be noted that even though the processing time at the edge layer (5 s) is selected four times longer than that of the fog layer (1 s), it is beneficial to process data at the edge layer and transfer only relevant data to higher layers during the peak hours (dawn and dusk) as done in the hybrid processing, especially in WSNs with low power communication protocol (e.g., ZigBee), since the packet latency and the transmission time are the main contributors in the notification time.

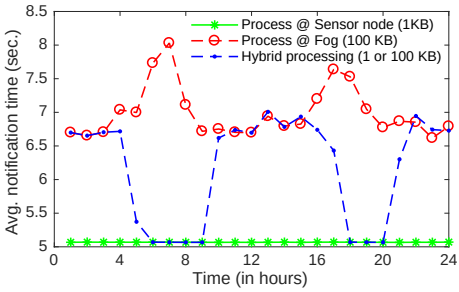

Figure 5.20.: Average notification time in seconds to alert human/vehicles using simulation results [96].

Finally, Fig. 5.21 illustrates the normalized fraction of dropped requests due to congestion of cellular and FTTH traffic with and without machine learning-based traffic prediction and bandwidth allocation, i.e., dynamic bandwidth allocation (DBA) and static bandwidth allocation (SBA), respectively, in the OFDM-PON. Without machine learning (i.e., in SBA), the bandwidth allocated to each ONU is statically divided among the three classes of traffic based on the proportion of their long-term average traffic per day. With machine learning (i.e., in DBA), the system divides bandwidth based on the predicted traffic for every hour, thus it exploits the dynamic assignment of subcarriers. Furthermore, the animal traffic is very low as compare to the cellular and the FTTH, thus the animal (class-3) packets are stored in enough memory when they could not be sent to the central office immediately. Thus, there is no dropped packets for the class-3 traffic at ONUs/BSs.

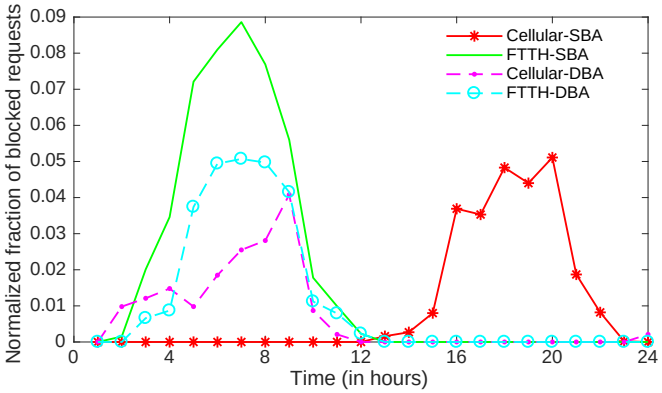

Figure 5.21.: Fraction of blocked requests of cellular and FTTH traffic [96].

The normalized fraction of dropped requests of a class in an hour

is calculated as the ratio of blocked requests in an hour and the maximum requests generated per hour during a day. Fig. 5.21 shows that the fraction of dropped connections under the SBA policy at any time for a class (FTTH or cellular) is proportional to the traffic (or mean arrival rate) of that class. On the other hand, we also see that AI-enabled dynamic management of optical resources, i.e., DBA, decreases the fraction of dropped connections during the peak hours for both traffic classes by assigning proportional bandwidth required for the traffic classes, thus effectively utilizing the statistical multiplexing gain. More importantly, the DBA gain for one class (FTTH) could come at the cost of increasing the fraction of dropped connections for another class (cellular) during a time interval, and it depends on the accuracy of predicted traffic. This can be seen in the first 12 hours of a day in Fig. 5.21, where the fraction of dropped connections of FTTH (cellular) traffic is decreased (increased) in the DBA scheme as compare to the SBA scheme. The reason is that the DBA allocates (reserves) proportionately higher ONUs' upstream bandwidth for the FTTH traffic in the first half of the day than the cellular traffic, since the predicted FTTH traffic is higher than the predicted cellular traffic in the first 12 hours (see Fig. 5.18).

5.5. Summary

This chapter outlined some generic use cases of AI and emerging challenges in optical transmission and networking. Then, we presented two specific scenarios of resource resource (re)allocation in optical datacenter networks, and in fiber-wireless networks, where machine learning algorithms was shown to be useful for prediction, classification and monitoring activities, as well as for decision making. In the resource (re)allocation in optical datacenters, we predicted lognormally-distributed mean holding time information of traffic classes, and estimated residual lifetimes of connections to

(re)allocate spectrum resources efficiently. Furthermore, we utilized fiber-wireless networks to build an early warning system framework, where different classes of traffic demands (animal, cellular and fiber-to-the-home) are predicted to decide whether to classify monitored animal data at sensors or at base stations, and when to transmit processed or unprocessed data to base stations in order to generate warning alert signals, if necessary, and to alert passing vehicles or humans.

6

Conclusion

Spectrum fragmentation is inevitable in EONs due to the spectrum contiguity and spectrum continuity constraints. Thus, this thesis presented the first exact Markov model for the modeling of κ-shortest paths routing and two different spectrum allocation schemes (random-fit and first-fit) toward computing the connection blocking probabilities in EONs. The exact model is accurate, albeit limited to very small scale networks due to its complexity. Nevertheless, it paves the way towards developing approximate blocking models. To this end, we used a known circuit-switched approach with the reduced load approximation and independence link assumption. The approximate models utilized various load-independent approximations for computing the probability of acceptance of a request in a given route occupancy, as well as a load-dependent approximation taking into account the slice occupancy correlation on a link. The approximation analysis advanced the state-of-the-art, and shown to be more accurate than previous approximations for computing blocking probabilities in EONs in polynomial time computation.

This thesis extended the exact and approximate blocking models towards modeling the spectrum defragmentation (DF) approach, which reconfigures exiting connections in order to reduce overall blocking. First, we proposed three different DF models in an elastic

optical link, where DF is triggered reactively, proactively, and by a hybrid approach with acceptance of delayed requests. Although these models show the benefit of DF, the blocking reduction is shown to be limited due to two major assumptions: during a DF process, all arriving requests are blocked and services of all connections are interrupted. Thus, we proposed an exact Markov model for a small scale network and approximate models which relaxed these two assumptions, and show individual blocking parts: fragmentation blocking and resource blocking in EONs with or without DF. Analytical and simulation results show that fragmentation blocking is the major source of blocking up to moderate loads in large networks irrespective of the spectrum allocation policies. Interestingly, we showed that a fraction of blocking that can not be eliminated in a mesh network topology even after using an optimal DF approach.

As spectrum fragmentation occurs in spectral, time and spatial domains, and analytical modeling of an adaptive spectrum allocation scheme is complex, we proposed a holding-time-aware resource allocation heuristic, which considers fragmentation factor in assigning resources to new connection requests. The simulation results showed that the proposed resource allocation scheme outperforms the first-fit spectrum allocation scheme which is shown to be simple yet very effective in reducing blocking probability in EONs. Furthermore, we used the holding times and remaining lifetimes of connections during the connection reconfiguration phase as well in the form of a proactive and a reactive DF schemes. Results showed that the reactive DF with acceptance of requests which could not be accepted without DF is the most beneficial among all proposed approaches. However, the complexity of this approach is non-polynomial in time, as compared to the polynomial complexity of the proactive DF approach.

Analytical RSA models are important for the network dimensioning problem and they can be used for accurate and quick forecasting of the resources' requirements to meet service level agreements at

a given network load. However, these models, including ours, generally analyze simple, yet complex to model, resource (re)allocation schemes. The efficient algorithms without adaptive features, on the other hand, can not learn non-linear characteristics of the dynamic network environment. Therefore, we envisioned the role of machine learning techniques in managing resources in optical networks. To this end, we first used machine learning for traffic prediction and (re)allocation of just enough resources to time-varying traffic in optical data center networks. The accurate prediction of heavy-tailed distributed traffic, and efficient utilization of residual lives of connections with unknown holding times led to minimizing connection blocking. The second use-case showed that an early warning system can be developed by incorporating machine learning techniques. We showed that the processing of sensor data at the edge devices can reduce energy consumption significantly.

Open Issues

In the stochastic analyses part, we observed that the elasticity in resource provisioning with spectrum continuity and contiguity constraints and resulted fragmentation in elastic optical networks are major challenges in modeling of the routing and spectrum allocation problem. The complexity increases when κ-shortest paths per node-pair are considered. The accuracy of models presented in this thesis and earlier degrades with increasing link capacity and number of paths. There are a number of possible reasons for the inaccuracy, including link-independence model inaccuracy, round-off errors, and inaccuracy with probability of acceptance approximations. As stochastic analyses will continue attracting researchers, the thesis advanced the state-of-the-art and showed the next steps, but also left some open issues and challenges, most notably the inter-dependency of network link loads, computing the probability of acceptance of

requests on a route accurately without increasing the complexity for practical reason. Furthermore, the thesis presented areas of application of learning-based algorithms in optical networks, and challenges, for example, monitoring multi-vendor devices in multiple network domains, extracting useful features from large collected data, etc. Whether it is mitigating spectrum fragmentation or monitoring of network parameters or attacks, (un)supervised learning methods require training examples, which is a challenge. Especially, models not trained with real examples will be hard to use in real networks. Therefore, a promising method that is emerging or even utilized for sometime now is reinforcement learning, which learns an optimal (or nearly optimal) policy from a series of reinforcements (rewards) or punishments received from its interaction with the environment. A possible use case in the line of the work presented in this thesis will be to apply Markov Decision Process as a reinforcement learning method to provisioning resources in a non-greedy fashion, that is applying policy-based treatment to different traffic classes and avoiding some network states which would lead to deadlock or fragmentation states that can not be defragmented.

Appendices

A. Derivation of total, Non-blocking and Blocking Link-states for the RF Policy

In this Appendix, we derive analytical expressions for computing the number of exact states ($|\Omega_S(x)|$) on a link with r traversing routes, the number of non-blocking exact link-states ($|\mathbb{NB}(x,k)|$), and the number of fragmented states ($|\mathbb{FI}(x,k)|$). Without loss of generality, let us assume that there are r number of different OD routes ($o = 1,\ldots,\mathrm{r}$) which traverse a link under consideration. For a given microstate $x, 0 \leqslant x \leqslant C$, where the number of empty (free) slices $E(x) = C - x$, we can find connection patterns or macrostates (i.e., connections per class per route) $\boldsymbol{n} = (n_1^1,\ldots,n_K^1,\ldots,n_1^{\mathrm{r}},\ldots,n_K^{\mathrm{r}})$ that satisfies $\boldsymbol{n}\cdot\boldsymbol{d}_{\mathrm{r}}^T = x$, where T is transpose, and $\boldsymbol{d}_{\mathrm{r}} = (d_1^1,\ldots,d_K^1,\ldots,d_1^{\mathrm{r}},\ldots,d_K^{\mathrm{r}})$ is an array with rK elements, and by definition for all OD pairs o, $d_k^o \equiv d_k$. Now, for each connection pattern $\boldsymbol{n} = (n_1^1,\ldots,n_K^1,\ldots,n_1^{\mathrm{r}},\ldots,n_K^{\mathrm{r}})$, the $E(x)$ empty slices can be distributed at $N(\boldsymbol{n})+1$ places (including the start, end, and in between each two connections), where the total number of connections $N(\boldsymbol{n}) = \sum_{k=1}^{K}\sum_{o=1}^{\mathrm{r}} n_k^o(\boldsymbol{n})$. Noting that there are $\frac{N(\boldsymbol{n})!}{\prod_{k=1}^{K}\prod_{o=1}^{\mathrm{r}} n_k^o(\boldsymbol{n})!}$ distinct permutations of connections in $\boldsymbol{n}$, and there are $\binom{E(x)+N(\boldsymbol{n})}{N(\boldsymbol{n})}$ different ways to distribute $E(x)$ empty slices at $N(\boldsymbol{n})+1$ places in each unique permutation of $\boldsymbol{n}$, the number of exact link-states with all connection patterns representing a microstate occupancy x is, thus, given by

$$|\Omega_S(x)| = \sum_{\boldsymbol{n}\in\Omega_S(x)} \frac{N(\boldsymbol{n})!}{\prod_{k=1}^{K}\prod_{o=1}^{\mathrm{r}} n_k^o(\boldsymbol{n})!} \times \binom{E(x)+N(\boldsymbol{n})}{N(\boldsymbol{n})}.$$

Importantly, only some of the exact link-states ($\mathbf{s}_i \in \Omega_S(x)$) are non-blocking states, as defined in (2.6). To compute the number of non-blocking exact link-states, let us solve the following equation for each permutation of the connection pattern for all $\boldsymbol{n} \in \Omega_S(x)$:

$$a_1 + a_2 + \cdots + a_{N(\boldsymbol{n})+1} = E(x), s.t., \exists i : a_i \geqslant d_k$$

Using the inclusion-exclusion principle (hint: consider an event $A_i = \{a_i \geqslant d_k\}, 1 \leqslant i \leqslant N(\boldsymbol{n}) + 1$ and find $|\cup_i A_i|$), the number of non-blocking exact states corresponding to each permutation of connections in $\boldsymbol{n} \in \Omega_S(x)$ can be given by

$$W(\boldsymbol{n}) = \sum_{i=1}^{N(\boldsymbol{n})+1} (-1)^{i+1} \binom{N(\boldsymbol{n}) + 1}{i} \binom{E(x) + N(\boldsymbol{n}) - id_k}{N(\boldsymbol{n})}.$$

Now, considering all permutations of $\boldsymbol{n}$, and adding all non-blocking states corresponding to each $\boldsymbol{n}$ belonging to the microstate x would result in the number of class k non-blocking exact states for a given microstate x, and it is given by

$$|\mathbb{NB}(x,k)| = \sum_{\boldsymbol{n} \in \Omega_S(x)} W(\boldsymbol{n}) \times \frac{N(\boldsymbol{n})!}{\prod_{k=1}^{K} \prod_{o=1}^{\mathrm{r}} n_k^o(\boldsymbol{n})!}.$$

Noting that for a class k request in any occupancy state x, $|\Omega_S(x)| = |\mathbb{NB}(x,k)| + |\mathbb{FI}(x,k)| + |\mathbb{RB}(x,k)|$, the number of fragmentation exact link-states in a microstate x is given by (2.10), and the number of resource blocking exact link-states is $|\mathbb{RB}(x,k)| = |\Omega_S(x)| - |\mathbb{NB}(x,k)| - |\mathbb{FI}(x,k)|$.

$$|\mathbb{FI}(x,k)| = \begin{cases} |\Omega_S(x)| - |\mathbb{NB}(x,k)|, & 0 \leqslant x \leqslant C - d_k \\ 0, & \text{otherwise.} \end{cases}$$

B. Uniform Approximation for Computing Probability of Acceptance

In this Appendix, we derive an analytical expression for computing approximate probability of acceptance $p_k(\mathbf{x}_r) \equiv p_k(x_{j_1}, x_{j_2}, \ldots, x_{j_h}) = Pr[Z_r \geq d_k | X_j = x_{j_1}, \ldots, X_{j_h} = x_{j_h}]$ on an h-hop route in an EON using a *Uniform* approach. Considering the independence link assumption, we further assume that *the occupancy of slices are independent and identically distributed in each link.* This means that total occupied slices are uniformly distributed, i.e., the spectrum patterns are formed by a single slice-demand with a given total occupancy, without considering the contiguous allocation of slices, and are not restricted to the spectrum patterns generated by a given classes of demands and spectrum allocation policy. Now, for a given occupancy of links on a route r, the probability that there are n continuous (but not necessarily contiguous) free slices on its route is obtained by the following recursive relationship [54, 55]:

$$\begin{aligned} g_n(x_{j_1}, x_{j_2}, \cdots, x_{j_h}) &= Pr[Z_r = n | X_{j_1} = x_{j_1}, \cdots, X_{j_h} = x_{j_h}] \\ &= \sum_{i=n}^{i^*} g_n(C-i, x_{j_h}) g_i(x_{j_1}, x_{j_2}, \cdots, x_{j_{l-1}}) \end{aligned} \tag{B-1}$$

where $i^* = min(C - x_{j_1}, C - x_{j_2}, \cdots, C - x_{j_{l-1}})$ and $g_n(x, y) = \binom{C-x}{n}\binom{x}{C-y-n}/\binom{C}{C-y}$.

Now, we could find the probability that the route r has equal or more than d_k free contiguous slices $\{Z_r \geq d_k\}$ across links on its route, given the link occupancy vector $\mathbf{x}_r$ and also there are exactly n continuous free slices on the route with $\{X_r = n\}$, i.e., $p_k^{Uni.}(\mathbf{x}_r, n) = Pr[Z_r \geq d_k | X_{j_1} = x_{j_1}, \cdots, X_{j_h} = x_{j_h}, X_r = n]$. Using the inclusion-exclusion principle, it can be given by

$$p_k^{App.Uni}(\mathbf{x}_r, n) = \frac{\sum_{i=1}^{C-n+1} (-1)^{i+1} \binom{C-n+1}{i} \binom{C-id_k}{C-n}}{\binom{C}{n}}. \tag{B-2}$$

The above equation seems to be independent of $\mathbf{x}_r$, but actually a factor which is a function of $\mathbf{x}_r$ is multiplied in both numerator and denominator, thus cancels the effect of $\mathbf{x}_r$. Thus, $p_k^{Uni.}(\mathbf{x}_r) = Pr[Z_r \geq d_k | X_{j_1} = x_{j_1}, \cdots, X_{j_h} = x_{j_h}]$ can be given as follows.

$$p_k^{App.Uni}(\mathbf{x}_r) = \sum_{n=d_k}^{min(C-\mathbf{x}_r)} p_k(\mathbf{x}_r, n) g_n(\mathbf{x}_r) \tag{B-3}$$

Under the *Uniform* approximation, the probability of acceptance in EONs with SC can easily be given by using Eq. (2.18). Noting that the spectrum patterns are assumed to be created by a single slice demand in the *Uniform* approximation. Thus, the probability that a link j in state x_j have equal or more than d_k free consecutive slices can be given by the ratio of non-blocking and total exact states in x_j as follows, which uses $\mathbf{n} = (n_1) = (x_j)$ and $N(\mathbf{n}) = x_j$ in Eqs. (2.9) and (2.8).

$$p_{k,sc}^{App.Uni}(x_j) = \frac{\sum_{i=1}^{x_j+1} (-1)^{i+1} \binom{x_j+1}{i} \binom{C-id_k}{x_j}}{\binom{C}{x_j}} \tag{B-4}$$

Finally, the probability of acceptance of a request with demand d_k on a route $r(o)$ with h-hops in an EON with SC can be obtain by multiplying link acceptance probabilities on the route $r(o)$, as $\prod_{\forall j \in r(o)} p_{k,sc}^{App.Uni}(x_j)$.

Bibliography

[1] V. López and L. Velasco, *Elastic Optical Networks: Architectures, Technologies, and Control.* Springer, 2016.

[2] N. Sambo, P. Castoldi, A. D'Errico, E. Riccardi, A. Pagano, M. S. Moreolo, J. M. Fabrega, D. Rafique, A. Napoli, S. Frigerio *et al.*, "Next generation sliceable bandwidth variable transponders," *Communications Magazine, IEEE*, vol. 53, no. 2, pp. 163–171, 2015.

[3] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, and S. Matsuoka, "Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies," *Communications Magazine, IEEE*, vol. 47, no. 11, pp. 66–73, 2009.

[4] O. Gerstel, M. Jinno, A. Lord, and S. B. Yoo, "Elastic optical networking: A new dawn for the optical layer?" *Communications Magazine, IEEE*, vol. 50, no. 2, pp. s12–s20, 2012.

[5] D. Richardson, J. Fini, and L. Nelson, "Space-division multiplexing in optical fibres," *Nature Photonics*, vol. 7, no. 5, p. 354, 2013.

[6] W. Shi, Z. Zhu, M. Zhang, and N. Ansari, "On the effect of bandwidth fragmentation on blocking probability in elastic

optical networks," *Communications, IEEE Transactions on*, vol. 61, no. 7, pp. 2970–2978, 2013.

[7] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Time-varying spectrum allocation policies and blocking analysis in flexible optical networks," *Selected Areas in Communications, IEEE Journal on*, vol. 31, no. 1, pp. 13–25, 2013.

[8] Y. Wang, X. Cao, and Y. Pan, "A study of the routing and spectrum allocation in spectrum-sliced elastic optical path networks," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 1503–1511.

[9] A. Sridharan and K. N. Sivarajan, "Blocking in all-optical networks," *IEEE/ACM transactions on networking*, vol. 12, no. 2, pp. 384–397, 2004.

[10] X. Chen, Y. Zhong, and A. Jukan, "Multipath routing in elastic optical networks with distance-adaptive modulation formats," in *Communications (ICC), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3915–3920.

[11] X. Chen, A. Jukan, and A. Gumaste, "Multipath defragmentation: Achieving better spectral efficiency in elastic optical path networks," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 390–394.

[12] Z. Zhu, W. Lu, L. Zhang, and N. Ansari, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," *Journal of Lightwave Technology*, vol. 31, no. 1, pp. 15–22, 2013.

[13] J. de Santi, A. C. Drummond, N. L. da Fonseca, X. Chen, and A. Jukan, "Leveraging multipath routing and traffic grooming

for an efficient load balancing in optical networks," in *Communications (ICC), 2012 IEEE International Conference on*. IEEE, 2012, pp. 2989–2993.

[14] S. K. Singh, T. Das, and A. Jukan, "A survey on internet multipath routing and provisioning," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2157–2175, 2015.

[15] K. Kayano, S. Yamaoka, Y. Mori, H. Hasegawa, and K.-i. Sato, "Highly dense elastic optical networks enabled by grouped routing with distance-adaptive modulation," *IEEE Photonics Technology Letters*, 2019.

[16] Y. Kishi, N. Kitsuwan, H. Ito, B. C. Chatterjee, and E. Oki, "Modulation-adaptive link-disjoint path selection model for 1+ 1 protected elastic optical networks," *IEEE Access*, 2019.

[17] X. Li, L. Zhang, Y. Tang, T. Gao, Y. Zhang, and S. Huang, "On-demand routing, modulation level and spectrum allocation (od-rmsa) for multicast service aggregation in elastic optical networks," *Optics express*, vol. 26, no. 19, pp. 24 506–24 530, 2018.

[18] L. R. Costa and A. C. Drummond, "New distance-adaptive modulation scheme for elastic optical networks," *IEEE Communications Letters*, vol. 21, no. 2, pp. 282–285, Feb 2017.

[19] M. S. Johnstone and P. R. Wilson, "The memory fragmentation problem: solved?" in *ACM SIGPLAN Notices*, vol. 34, no. 3. ACM, 1998, pp. 26–36.

[20] Y. Yu, J. Zhang, Y. Zhao, X. Cao, X. Lin, and W. Gu, "The first single-link exact model for performance analysis of flexible grid wdm networks," in *National Fiber Optic Engineers Conference*. Optical Society of America, 2013, pp. JW2A–68.

[21] L. Peng, C.-H. Youn, and C. Qiao, "Theoretical analyses of lightpath blocking performance in co-ofdm optical networks with/without spectrum conversion," *IEEE Communications Letters*, vol. 17, no. 4, pp. 789–792, 2013.

[22] H. Beyranvand, M. Maier, and J. Salehi, "An analytical framework for the performance evaluation of node-and network-wise operation scenarios in elastic optical networks," *Communications, IEEE Transactions on*, vol. 62, no. 5, pp. 1621–1633, 2014.

[23] R. R. Reyes and T. Bauschert, "Reward-based online routing and spectrum assignment in flex-grid optical networks," in *Telecommunications Network Strategy and Planning Symposium (Networks), 2016 17th International*. IEEE, 2016, pp. 101–108.

[24] Y. Sone, A. Hirano, A. Kadohata, M. Jinno, and O. Ishida, "Routing and spectrum assignment algorithm maximizes spectrum utilization in optical networks," in *European Conference and Exposition on Optical Communications*. Optical Society of America, 2011, pp. Mo–1.

[25] Y. Yin, Z. Zhu, S. Yoo *et al.*, "Fragmentation-aware routing, modulation and spectrum assignment algorithms in elastic optical networks," in *Optical Fiber Communication Conference*. Optical Society of America, 2013, pp. OW3A–5.

[26] X. Chen, S. Ma, B. Guo, Y. Wang, J. Li, Z. Chen, and Y. He, "A novel fragmentation-aware spectrum allocation algorithm in flexible bandwidth optical networks," *Optical Switching and Networking*, vol. 12, pp. 14–23, 2014.

[27] W. Lu and Z. Zhu, "Dynamic service provisioning of advance

reservation requests in elastic optical networks," *Journal of Lightwave Technology*, vol. 31, no. 10, pp. 1621–1627, 2013.

[28] H. Chen, Y. Zhao, J. Zhang, R. He, W. Wang, J. Wu, Y. Wang, Y. Ji, H. Zheng, Y. Lin *et al.*, "Time-spectrum consecutiveness based scheduling with advance reservation in elastic optical networks," *Communications Letters, IEEE*, vol. 19, no. 1, pp. 70–73, 2015.

[29] N. Wang, J. P. Jue, X. Wang, Q. Zhang, H. C. Cankaya, and M. Sekiya, "Holding-time-aware scheduling for immediate and advance reservation in elastic optical networks," in *Communications (ICC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5180–5185.

[30] B. C. Chatterjee, N. Sarma, and E. Oki, "Routing and spectrum allocation in elastic optical networks: A tutorial," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1776–1800, 2015.

[31] B. C. Chatterjee, S. Ba, and E. Oki, "Fragmentation problems and management approaches in elastic optical networks: A survey," *IEEE COMMUNICATIONS SURVEYS & TUTORIALS*, vol. 20, no. 1, 2018.

[32] M. Zhang, C. You, and Z. Zhu, "On the parallelization of spectrum defragmentation reconfigurations in elastic optical networks," *Networking, IEEE/ACM Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[33] A. N. Patel, P. N. Ji, J. P. Jue, and T. Wang, "Defragmentation of Transparent Flexible Optical WDM (FWDM) Networks," in *Optical Fiber Communication Conference*. Optical Society of America, 2011, p. OTuI8.

[34] R. Wang and B. Mukherjee, "Provisioning in elastic optical networks with non-disruptive defragmentation," *Journal of Lightwave Technology*, vol. 31, no. 15, pp. 2491–2500, 2013.

[35] X. Yu, J. Zhang, Y. Zhao, T. Peng, Y. Bai, D. Wang, and X. Lin, "Spectrum compactness based defragmentation in flexible bandwidth optical networks," in *National Fiber Optic Engineers Conference*. Optical Society of America, 2012, pp. JTh2A–35.

[36] S. K. Singh and A. Jukan, "Non-disruptive spectrum defragmentation with holding-time awareness in optical networks," in *Optical Network Design and Modeling (ONDM)*. IEEE, 2016.

[37] S. K. Singh, W. Bziuk, and A. Jukan, "Analytical performance modeling of spectrum defragmentation in elastic optical link networks," *Optical Switching and Networking*, vol. 24, pp. 25–38, 2017.

[38] H. Tode and Y. Hirota, "Routing, spectrum and core assignment for space division multiplexing elastic optical networks," in *Telecommunications Network Strategy and Planning Symposium (Networks), 2014 16th International*. IEEE, 2014, pp. 1–7.

[39] S. Ba, B. C. Chatterjee, S. Okamoto, N. Yamanaka, A. Fumagalli, and E. Oki, "Route partitioning scheme for elastic optical networks with hitless defragmentation," *Journal of Optical Communications and Networking*, vol. 8, no. 6, pp. 356–370, 2016.

[40] Y. Yin, K. Wen, D. J. Geisler, R. Liu, and S. Yoo, "Dynamic on-demand defragmentation in flexible bandwidth elastic op-

tical networks," *Optics Express*, vol. 20, no. 2, pp. 1798–1804, 2012.

[41] S. Shakya and X. Cao, "Spectral defragmentation in elastic optical path networks using independent sets," in *National Fiber Optic Engineers Conference*. Optical Society of America, 2013, pp. NTh1I–4.

[42] M. Zhang, C. You, H. Jiang, and Z. Zhu, "Dynamic and adaptive bandwidth defragmentation in spectrum-sliced elastic optical networks with time-varying traffic," *Journal of Lightwave Technology*, vol. 32, no. 5, pp. 1014–1023, 2014.

[43] S. K. Singh, W. Bziuk, and A. Jukan, "Defragmentation-as-a-Service (DaaS): How beneficial is it?" in *Optical Fiber Communication Conference*. Optical Society of America, 2016, pp. W2A–55.

[44] ——, "Balancing data security and blocking performance with spectrum randomization in optical networks," in *Global Communications Conference (GLOBECOM), 2016 IEEE*. IEEE, 2016, pp. 1–7.

[45] T. Takagi, H. Hasegawa, K.-i. Sato, Y. Sone, A. Hirano, and M. Jinno, "Disruption minimized spectrum defragmentation in elastic optical path networks that adopt distance adaptive modulation," in *European Conference and Exposition on Optical Communications*. Optical Society of America, 2011, pp. Mo–2.

[46] F. Cugini, F. Paolucci, G. Meloni, G. Berrettini, M. Secondini, F. Fresi, N. Sambo, L. Poti, and P. Castoldi, "Push-pull defragmentation without traffic disruption in flexible grid optical networks," *Journal of lightwave technology*, vol. 31, no. 1, pp. 125–133, 2013.

[47] Y. Takita, K. Tajima, T. Hashiguchi, and T. Katagiri, "Wavelength defragmentation for seamless service migration," *Journal of Optical Communications and Networking*, vol. 9, no. 2, pp. A154–A161, 2017.

[48] C. Kachris, K. Kanonakis, and I. Tomkos, "Optical interconnection networks in data centers: recent trends and future challenges," *Communications Magazine, IEEE*, vol. 51, no. 9, pp. 39–45, 2013.

[49] Y. Li, H. Liu, W. Yang, D. Hu, X. Wang, and W. Xu, "Predicting inter-data-center network traffic using elephant flow and sublink information," *IEEE Transactions on Network and Service Management*, vol. 13, no. 4, pp. 782–792, 2016.

[50] B. Kozicki, H. Takara, Y. Tsukishima, T. Yoshimatsu, T. Kobayashi, K. Yonenaga, and M. Jinno, "Optical path aggregation for 1-tb/s transmission in spectrum-sliced elastic optical path network," *IEEE Photonics Technology Letters*, vol. 22, no. 17, pp. 1315–1317, 2010.

[51] J. Kim, S. Yan, A. Fumagalli, E. Oki, and N. Yamanaka, "An analytical model of spectrum fragmentation in a two-service elastic optical link," in *Global Communications Conference (GLOBECOM), 2015 IEEE*. IEEE, 2015, pp. 1–6.

[52] A. Rosa, P. Wiatr, C. Cavdar, S. Carvalho, J. Costa, and L. Wosinska, "Statistical analysis of blocking probability and fragmentation based on markov modeling of elastic spectrum allocation on fiber link," *Optics Communications*, vol. 354, pp. 362–373, 2015.

[53] F. Ge and L. Tan, "Blocking performance approximation in flexi-grid networks," *Optical Fiber Technology*, vol. 32, pp. 58–65, 2016.

[54] A. Birman, "Computing approximate blocking probabilities for a class of all-optical networks," *IEEE Journal on Selected areas in Communications*, vol. 14, no. 5, pp. 852–857, 1996.

[55] K. Kuppuswamy and D. C. Lee, "An analytic approach to efficiently computing call blocking probabilities for multi-class wdm networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 17, no. 2, pp. 658–670, 2009.

[56] J. Kaufman, "Blocking in a shared resource environment," *IEEE Transactions on communications*, vol. 29, no. 10, pp. 1474–1481, 1981.

[57] E. Karasan and E. Ayanoglu, "Effects of wavelength routing and selection algorithms on wavelength conversion gain in wdm optical networks," *IEEE/ACM Transactions on Networking*, vol. 6, no. 2, pp. 186–196, April 1998.

[58] S. K. Singh and A. Jukan, "Computing exact and approximate blocking probabilities in elastic optical networks," *arXiv preprint arXiv:1706.07747*, 2018. [Online]. Available: https://arxiv.org/pdf/1706.07747.pdf

[59] L. Kleinrock, *Queueing systems, volume 1: theory*. John Wiley & Sons New York, 1975.

[60] S. K. Singh and A. Jukan, "Efficient spectrum defragmentation with holding-time awareness in elastic optical networks," *Journal of Optical Communications and Networking*, vol. 9, no. 3, pp. B78–B89, 2017.

[61] C. C. Paige and M. A. Saunders, "Lsqr: An algorithm for sparse linear equations and sparse least squares," *ACM transactions on Mathematical Software*, vol. 8, no. 1, pp. 43–71, 1982.

[62] C. Chow and C. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE transactions on Information Theory*, vol. 14, no. 3, pp. 462–467, 1968.

[63] S.-P. Chung, A. Kashper, and K. W. Ross, "Computing approximate blocking probabilities for large loss networks with state-dependent routing," *IEEE/ACM Transactions on Networking (TON)*, vol. 1, no. 1, pp. 105–115, 1993.

[64] S. Fujii, Y. Hirota, H. Tode, and K. Murakami, "On-demand spectrum and core allocation for reducing crosstalk in multi-core fibers in elastic optical networks," *J. of Optical Communications and Networking*, vol. 6, no. 12, pp. 1059–1071, 2014.

[65] S. K. Singh, W. Bziuk, and A. Jukan, "A combined optical spectrum scrambling and defragmentation in multi-core fiber networks," in *Communications (ICC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1–6.

[66] A. Castro, L. Velasco, M. Ruiz, M. Klinkowski, J. P. FernáNdez-Palacios, and D. Careglio, "Dynamic routing and spectrum (re) allocation in future flexgrid optical networks," *Computer Networks*, vol. 56, no. 12, pp. 2869–2883, 2012.

[67] R. Proietti, C. Qin, B. Guan, Y. Yin, R. P. Scott, R. Yu, and S. Yoo, "Rapid and complete hitless defragmentation method using a coherent rx lo with fast wavelength tracking in elastic optical networks," *Optics express*, vol. 20, no. 24, pp. 26 958–26 968, 2012.

[68] S. K. Singh and A. Jukan, "Computing blocking probabilities in elastic optical networks with spectrum defragmentation," in *IEEE International conference on Computer Communications (INFOCOM)*. IEEE, 2019, pp. 1–9.

[69] B. C. Chatterjee, S. Ba, and E. Oki, "Fragmentation problems and management approaches in elastic optical networks: a survey," *IEEE Communications Surveys & Tutorials*, 2017.

[70] M. Tornatore, A. Baruffaldi, H. Zhu, B. Mukherjee, and A. Pattavina, "Holding-time-aware dynamic traffic grooming," *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 3, pp. 28–35, 2008.

[71] D. Amar, E. Le Rouzic, N. Brochier, J.-L. Auge, C. Lepers, and N. Perrot, "Spectrum fragmentation issue in flexible optical networks: analysis and good practices," *Photonic Network Communications*, vol. 29, no. 3, pp. 230–243, 2015.

[72] S. K. Singh and A. Jukan, "Holding-Time Information (HTI): When to Use it?" in *Optical Fiber Communication Conference*. Optical Society of America, 2017.

[73] A. Sharieh, W. Al Rawagepfeh, M. Mahafzah, and A. Al Dahamsheh, "An algorithm for finding maximum independent set in a graph," *European Journal of Scientific Research*, vol. 23, no. 4, pp. 586–596, 2008.

[74] M. Salehie and L. Tahvildari, "Self-adaptive software: Landscape and research challenges," *ACM transactions on autonomous and adaptive systems (TAAS)*, vol. 4, no. 2, p. 14, 2009.

[75] J. Mata, I. de Miguel, R. J. Durán, N. Merayo, S. K. Singh, A. Jukan, and M. Chamania, "Artificial intelligence (ai) methods in optical networks: A comprehensive survey," *Optical Switching and Networking*, vol. 24, pp. 25–38, 2018.

[76] F. Musumeci, C. Rottondi, A. Nag, I. Macaluso, D. Zibar, M. Ruffini, and M. Tornatore, "An overview on application of

machine learning techniques in optical networks," *IEEE Communications Surveys Tutorials*, 2018.

[77] J. Thrane, J. Wass, M. Piels, J. C. Diniz, R. Jones, and D. Zibar, "Machine learning techniques for optical performance monitoring from directly detected pdm-qam signals," *Journal of Lightwave Technology*, vol. 35, no. 4, pp. 868–875, 2017.

[78] J. Mata, I. de Miguel, R. J. Durán, N. Merayo, S. K. Singh, A. Jukan, and M. Chamania, "Application of artificial intelligence techniques in optical networks," in *2018 IEEE Photonics Society Summer Topical Meeting Series (SUM)*. IEEE, 2018, pp. 35–36.

[79] B. Karanov, M. Chagnon, F. Thouin, T. A. Eriksson, H. Bülow, D. Lavery, P. Bayvel, and L. Schmalen, "End-to-end deep learning of optical fiber communications," *Journal of Lightwave Technology*, vol. 36, no. 20, pp. 4843–4855, 2018.

[80] I. de Miguel, R. J. Durán, T. Jiménez, N. Fernández, J. C. Aguado, R. M. Lorenzo, A. Caballero, I. T. Monroy, Y. Ye, A. Tymecki *et al.*, "Cognitive dynamic optical networks," *Journal of Optical Communications and Networking*, vol. 5, no. 10, pp. A107–A118, 2013.

[81] M. Bensalem, S. K. Singh, and A. Jukan, "On detecting and preventing jamming attacks with machine learning in optical networks," in *under submission in IEEE GLOBECOM*. IEEE, 2019.

[82] T. Benson, A. Akella, and D. A. Maltz, "Network traffic characteristics of data centers in the wild," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*. ACM, 2010, pp. 267–280.

[83] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken, "The nature of data center traffic: measurements & analysis," in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*. ACM, 2009, pp. 202–208.

[84] A. Roy, H. Zeng, J. Bagga, G. Porter, and A. C. Snoeren, "Inside the social network's (datacenter) network," in *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 4. ACM, 2015, pp. 123–137.

[85] Y. Chen, S. Jain, V. K. Adhikari, Z.-L. Zhang, and K. Xu, "A first look at inter-data center traffic characteristics via yahoo! datasets," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 1620–1628.

[86] M. Klinkowski, M. Ruiz, L. Velasco, D. Careglio, V. Lopez, and J. Comellas, "Elastic spectrum allocation for time-varying traffic in flexgrid optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 1, pp. 26–38, 2013.

[87] P. S. Khodashenas, J. Comellas, S. Spadaro, J. Perelló, and G. Junyent, "Using spectrum fragmentation to better allocate time-varying connections in elastic optical networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 6, no. 5, pp. 433–440, 2014.

[88] S. Shakya, X. Cao, Z. Ye, and C. Qiao, "Spectrum allocation in spectrum-sliced elastic optical path networks using traffic prediction," *Photonic Network Communications*, vol. 30, no. 1, pp. 131–142, Aug 2015. [Online]. Available: https://doi.org/10.1007/s11107-015-0489-z

[89] Y. Zong, C. Yu, Y. Liu, Q. Zhang, Y. Sun, and L. Guo, "Time-dependent load-balancing service degradation in optical data

center networks," *Photonic Network Communications*, pp. 1–11, 2017.

[90] S. K. Singh and A. Jukan, "Machine-learning-based prediction for resource (re)allocation in optical data center networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, no. 10, pp. D12–D28, Oct 2018.

[91] F. Guess and F. Proschan, "12 mean residual life: Theory and applications," *Handbook of statistics*, vol. 7, pp. 215–224, 1988.

[92] K. Govil and K. Aggarwal, "Mean residual life function for normal, gamma and lognormal densities," *Reliability Engineering*, vol. 5, no. 1, pp. 47–51, 1983.

[93] Y. Xiong, J. Shi, Y. Yang, Y. Lv, and G. Rouskas, "Lightpath management in sdn-based elastic optical networks with power consumption considerations," *Journal of Lightwave Technology*, 2017.

[94] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[95] C. Ranaweera, E. Wong, C. Lim, and A. Nirmalathas, "Next generation optical-wireless converged network architectures," *IEEE Network*, vol. 26, no. 2, 2012.

[96] S. Singh, F. Carpio, and A. Jukan, "Improving animal-human cohabitation with machine learning in fiber-wireless networks," *Journal of Sensor and Actuator Networks*, vol. 7, no. 3, p. 35, 2018.

[97] K. Habel *et al.*, "100G OFDM-PON for Converged 5G Networks: From Concept to Realtime Prototype," in *Optical Fiber Communication Conference*. Optical Society of America, 2017.

[98] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[99] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[100] https://www.tensorflow.org/mobile/tflite/, "Tensorflow."

[101] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi, and A. P. Sheth, "Machine learning for internet of things data analysis: A survey," *Digital Communications and Networks*, 2017.

[102] P. Suman, P. Gupta, P. B. Kassey, N. Saxena, Y. Choudhary, V. Singh, and M. Radhakrishna, "Identification of trespasser from the signatures of buried single mode fiber optic sensor cable," in *India Conference (INDICON), 2015 Annual IEEE*. IEEE, 2015, pp. 1–6.

[103] Riverbed, "Riverbed Modeler Academic Edition release 17.5 A PL7," 2014. [Online]. Available: https://splash.riverbed.com/community/product-lines/steelcentral/university-support-center/blog/2014/06/11/riverbed-modeler-academic-edition-release

Acronyms

AI	Artificial Intelligence
ANN	Artificial Neural Networks
BER	Bit Error Rate
BSs	Base Stations
BP	Blocking Probability
CNN	Convolutional Neural Network
CTMC	Continuous Time Markov Chain
DBP	Defragmentation Blocking Probability
DC	Data Center
DF	Defragmentation
DP-QPSK	Dual Polarized Quadrature Phase Shift Keying
DT	Decision Tree
DWDM	Dense Wavelength Division Multiplexing
EES	Equiprobable Exact States
EDFA	Erbium-Doped Fiber Amplifier
EOL	Elastic Optical Link
EON	Elastic Optical Network
EOV	Equiprobable Occupancy Vector
FBP	Fragmentation Blocking Probability
FF	First Fit
FiWi	Fiber-Wireless
GBE	Global Balance Equation
HTA	Hooding-Time-Aware

HTI	Holding Time Information
ILP	Integer Linear Programming
IoT	Internet-of-Things
ITU-T	International Telecommunication Union-Telecom
KNN	K-Nearest Neighbors
LP	Lightpath
LR	Logistic Regression
ML	Machine Learning
MRL	Mean Residual Life
NB	Naive Bayesian
OD	Origin Destination
ODCN	Optical Data Center Network
OFDM	Orthogonal Frequency Division Multiplexing
OLT	Optical Line Terminal
ONUs	Optical Network Units
OSNR	Optical Signal-to-Noise Ratio
OTN	Optical Transport Networks
OXC	Optical Cross Connect
PON	Passive Optical Network
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
RBP	Resource Blocking Probability
RF	Random Fit
RFC	Request for Comments
ROADM	Reconfigurable Optical Add-Drop Multiplexer
RSA	Routing and Spectrum Allocation
RWA	Routing and Wavelength Assignment
SA	Spectrum Allocation
SC	Spectrum Conversion

SOC	Slice Occupancy Correlation
SP	Spectrum Path
SVM	Support Vector Machine
WDM	Wavelength Division Multiplexing
WSN	Wireless Sensor Network